

**IMPLEMENTASI SALES FORCE AUTOMATION BERBASIS
ANDROID PADA USAHA KERIPIK BABY FISH**

SKRIPSI



**Disusun Oleh :
TRI WAHYU PAMUNGKAS PRIBADI
10.18.018**

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2014

**MILIK
PERPUSTAKAAN
ITN MALANG**

ИЛИ ИУЛАНС
БЕКЪУСТАКЪАИ
ИЮЛІК

2019
ИРИШІІІ ТЕКНОЛОГИИ ИУФІОИУТ ИРИШІІІ
БАНКІІІІІ ТЕКНОЛОГИИ ИУФІОИУТ
ИНОУИИ ПІІІІІ ТЕКНИК ИІІІІІІІІІІІІІ І-І

10.10.2019
ІІІ ІУІІІІІ ІІІІІІІІІІІІ ІІІІІІІІ
ИІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ

ИІІІІІІ

ИІІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ
ИІІІІІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ ІІІІІ

LEMBAR PERSETUJUAN DAN PENGESAHAN
IMPLEMENTASI SALES FORCE AUTOMATION BERBASIS
ANDROID PADA USAHA KERIPIK BABY FISH

SKRIPSI

Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna mencapai Gelar Sarjana Komputer Strata Satu (S-1)

Disusun Oleh :
TRI WAHYU PAMUNGKAS PRIBADI
10.18.018

Diperiksa dan Disetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Joseph Dedy Irawan, ST., MT.
NIP. 197404162005011002

Suryo Adi Wibowo, ST., MT
NIP.P. 1031000438

Program Studi Teknik Informatika S-1
Ketua

Joseph Dedy Irawan, ST., MT
NIP. 197404162005011002

PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2014

LEMBAR KEASLIAN
PERNYATAAN KEASLIAN SKRIPSI

Saya yang bertanda tangan di bawah ini:

Nama : Tri Wahyu Pamungkas Pribadi
NIM : 10.18.018
Program Studi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Menyatakan dengan sesungguhnya bahwa Skripsi saya yang berjudul:

**“IMPLEMENTASI SALES FORCE AUTOMATION BERBASIS
ANDROID PADA USAHA KERIPIK BABY FISH”**

Adalah Skripsi saya sendiri bukan duplikat serta mengutip atau menyadur seluruhnya karya orang lain kecuali dari sumber aslinya.

Malang, 13 Februari 2014

Yang membuat pernyataan



Tri Wahyu Pamungkas Pribadi

IMPLEMENTASI SALES FORCE AUTOMATION BERBASIS ANDROID PADA USAHA KERIPIK BABY FISH

TRI WAHYU PAMUNGKAS PRIBADI

Program Studi Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Jl. Raya Karanglo Km. 2 Tasikmadu-Malang
Email : why755@gmail.com

**Dosen Pembimbing: 1. Joseph Dedy Irawan, ST., MT.
2. Suryo Adi Wibowo, ST., MT.**

Abstraksi

Sales Force Automation (SFA) merupakan jenis program yang mengotomasi pekerjaan atau tugas-tugas bisnis seperti mengontrol inventori, proses penjualan, pelacakan interaksi pelanggan dan menganalisa proyeksi penjualan beserta kinerjanya.

Perancangan sistem sales force automation terdiri dari 2 bagian yaitu aplikasi berbasis mobile android yang digunakan oleh retail dan aplikasi berbasis web sebagai server yang digunakan oleh supplier. Pada aplikasi mobile digunakan untuk melakukan pemesanan keripik, sedangkan pada bagian server memberikan informasi produk, promosi dan histori penjualan pada aplikasi mobile.

Hasil yang diperoleh dari penerapan sales force automation adalah memberi kemudahan bagi supplier untuk mengetahui daftar stok keripik yang tersedia pada setiap retail karena secara otomatis daftar stok produk akan berkurang sesuai jumlah permintaan produk yang dipesan dan pada jarak 2 hari daftar stok akan normal kembali pada jumlah semula. Pada bagian pelanggan, aplikasi dapat berjalan 100% dari segi tampilan dan 100% dari segi fungsi pada perangkat mobile dengan versi android dan pixel yang berbeda-beda yaitu 240 x 320, 320 x 480, 480 x 800, 480 x 854, 540 x 960 dan 1024 x 600 baik dalam posisi landscape maupun potrait dengan berbagai jenis operating system dari versi Froyo (2.2.1) sampai versi Jelly Bean (4.1.2).

Kata kunci : Sales Force Automation, Aplikasi Mobile, Android.

KATA PENGANTAR

Puji syukur kehadiran Allah Yang Maha Kuasa, karena telah memberikan rahmat dan hidayah-Nya sehingga penyusun dapat menyelesaikan skripsi dengan judul *IMPLEMENTASI SALES FORCE AUTOMATION BERBASIS ANDROID PADA USAHA KERIPIK BABY FISH* sesuai waktu yang telah ditentukan.

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada penyusunan skripsi ini kami mengucapkan banyak terima kasih sebesar-besarnya kepada :

1. Bapak Priyo Admadji dan Ibu Wahyu Indarwati, yang merupakan orang tua dan pendukung utama dari segi moril maupun materil.
2. Ir. Soeparno Djiwo, MT, selaku Rektor Institut Teknologi Nasional Malang.
3. Ir. Anang Subardi, MT, selaku Dekan Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.
4. Joseph Dedy Irawan, ST, MT, selaku Ketua Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
5. Sonny Prasetio, ST, MT, selaku Sekertaris Program Studi Teknik Informatika, Institut Teknologi Nasional Malang.
6. Joseph Dedy Irawan, ST, MT, selaku Dosen Pembimbing I, yang selalu memberikan masukan.
7. Suryo Adi Wibowo, ST., MT, selaku Dosen Pembimbing II, yang selalu memberikan masukan.
8. Semua Dosen Program Studi Teknik Informatika yang telah membantu dalam penulisan dan masukan.
9. Rekan-rekan yang telah memberi bantuan dalam menyusun penulisan skripsi ini.

Penulis menyadari bahwa penulisan skripsi ini masih jauh dari sempurna, oleh karena itu penulis mengharapkan kritik dan saran dari pembaca. Semoga skripsi ini bisa bermanfaat bagi pembaca.

Malang, Februari 2014

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN DAN PENGESAHAN	ii
LEMBAR KEASLIAN	iii
ABSTRAKSI	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Manfaat	2
1.4 Tujuan	2
1.5 Batasan Masalah	2
1.6 Metodologi Penelitian	3
1.6.1 Tahap Pengumpulan Data	3
1.6.2 Tahap Pembuatan	4
1.7 Sistematika Penulisan	6
BAB II LANDASAN TEORI	8
2.1 Baby Fish	8
2.1.1 Visi dan Misi	9
2.1.2 Logo <i>Baby Fish</i>	9
2.2 Sales Force Automation (SFA).....	10
2.3 Aplikasi <i>Mobile</i>	11
2.4 Android	12
2.5 PHP	14
2.6 MySQL	15
2.7 XAMPP	15
2.8 Eclipse	16
2.9 Java	18
2.10 Java Development Kit (JDK)	19
2.11 Software Development Kit (SDK)	19
2.12 Android Virtual Device (AVD)	20
BAB III ANALISIS DAN PERANCANGAN SISTEM	21
3.1 Analisis Sistem	21
3.1.1 Analisis Operasi Penjualan Konvensional	21
3.1.2 Analisis Operasi Sales Force Automation	22
3.1.3 Analisis Kebutuhan Data	24
3.1.4 Analisis Kebutuhan Perangkat Lunak	24

3.2 Perancangan Sistem	25
3.2.1 Rancangan Struktur Navigasi	26
3.2.2 Rancangan Diagram Alir	27
3.2.3 Rancangan Proses Pemesanan	29
3.2.4 Rancangan Proses Manual Notification	30
3.2.5 Flowchart Auto Notification Pada Proses Pemesanan	31
3.2.6 Rancangan Data Flow Diagram (DFD) Level 0	32
3.2.7 Rancangan Data Flow Diagram (DFD) Level 1	32
3.2.8 Entyty Relation Diagram	33
3.3 Desain Interface Sistem	34
3.3.1 Halaman Login Admin	34
3.3.2 Halaman Menu Administrator	35
3.3.3 Form Login Android Mobile	36
3.3.4 Form Registrasi Android Mobile	36
3.3.5 Tampilan Menu Beranda	37
3.3.6 Tampilan Menu Produk	37
3.3.7 Tampilan Menu Histori	38
3.3.8 Tampilan Menu Bantuan	38
BAB IV IMPLEMENTASI DAN PENGUJIAN	39
4.1 Implementasi Sistem	39
4.1.1 Bagian Administrasi	39
4.1.1.1 Tampilan Halaman Login	38
4.1.1.2 Tampilan Halaman Beranda	40
4.1.1.3 Tampilan Halaman Daftar Pelanggan	42
4.1.1.4 Tampilan Halaman Tambah Data Pelanggan	42
4.1.1.5 Tampilan Halaman Edit Data Pelanggan	43
4.1.1.6 Tampilan Halaman Daftar Ulang Tahun	43
4.1.1.7 Tampilan Halaman Edit Ulang Tahun	44
4.1.1.8 Tampilan Halaman Daftar Stok Retail	44
4.1.1.9 Tampilan Halaman Edit Stok Retail	45
4.1.1.10 Tampilan Halaman Daftar Karyawan	45
4.1.1.11 Tampilan Halaman Tambah Data Karyawan	46
4.1.1.12 Tampilan Halaman Edit Data Karyawan	46
4.1.1.13 Tampilan Halaman Daftar Produk	47
4.1.1.14 Tampilan Halaman Tambah Produk	47
4.1.1.15 Tampilan Halaman Edit Produk	48
4.1.1.16 Tampilan Halaman Daftar Pesan	48
4.1.1.17 Tampilan Halaman Edit Status Pesan	49
4.1.1.18 Tampilan Halaman Daftar Penjualan	49
4.1.1.19 Tampilan Halaman Tambah Data Penjualan	50
4.1.1.20 Tampilan Halaman Edit Data Penjualan	50
4.1.1.21 Tampilan Halaman Daftar Informasi	51
4.1.1.22 Tampilan Halaman Tambah Informasi	51
4.1.1.23 Tampilan Halaman Edit Infomasi	52
4.1.1.24 Tampilan Halaman Diagram Penjualan	52

4.1.1.25 Tampilan Halaman Statistik Penjualan	53
4.1.2 Bagian Pelanggan	53
4.1.2.1 Splash Screen	53
4.1.2.2 Form Login	54
4.1.2.3 Form Daftar Pelanggan	55
4.1.2.4 Menu Beranda	55
4.1.2.5 Menu Produk	56
4.1.2.6 Form Pemesanan	56
4.1.2.7 Menu Histori	57
4.1.2.9 Menu Bantuan	58
4.2 Pengujian	60
4.2.1 Pengujian Sistem	60
4.2.2 Pengujian Sistem Aplikasi Web Pada Beberapa Web Browser	62
4.2.3 Pengujian Sistem Aplikasi Android Pada Beberapa Telepon Genggam	63
4.2.4 Pengujian Sales Force Automation	65
 BAB V PENUTUP	 68
5.1 Kesimpulan	68
5.2 Saran	69
 DAFTAR PUSTAKA	 70
LAMPIRAN	71

DAFTAR GAMBAR

Gambar 1.1	<i>Model Waterfall</i>	5
Gambar 2.1	<i>Logo Usaha Keripik Baby Fish</i>	9
Gambar 3.1	<i>Flowchart Operasi Penjualan Konvensional</i>	21
Gambar 3.2	<i>Flowchart Operasi Sales Force Automation</i>	23
Gambar 3.3	<i>Arsitektur Sistem Pemesanan Produk</i>	25
Gambar 3.4	<i>Arsitektur Sistem Notifikasi Pembayaran</i>	25
Gambar 3.5	<i>Struktur Navigasi Pada Aplikasi Web</i>	26
Gambar 3.6	<i>Struktur Navigasi Pada Aplikasi Android</i>	27
Gambar 3.7	<i>Flowchart Pada Aplikasi Server</i>	28
Gambar 3.8	<i>Flowchart Pada Aplikasi Android</i>	29
Gambar 3.9	<i>Proses Sales Force Automation Pemesanan Keripik</i>	30
Gambar 3.10	<i>Proses Manual Notification</i>	31
Gambar 3.11	<i>Flowchart Auto Notification Pada Proses Pemesanan</i>	31
Gambar 3.12	<i>Data Flow Diagram Level 0</i>	32
Gambar 3.13	<i>Data Flow Diagram Level 1</i>	33
Gambar 3.14	<i>Entyty Relation Diagram</i>	34
Gambar 3.15	<i>Desain Prototipe Halaman Login Admin</i>	35
Gambar 3.16	<i>Desain Prototipe Halaman Menu Administrator</i>	35
Gambar 3.17	<i>Desain Prototipe Form Login Android Mobile</i>	36
Gambar 3.18	<i>Desain Prototipe Form Registrasi</i>	36
Gambar 3.19	<i>Desain Prototipe Android Mobile Menu Beranda</i>	37
Gambar 3.20	<i>Desain Prototipe Android Mobile Menu Produk</i>	37
Gambar 3.21	<i>Desain Prototipe Android Mobile Menu Histori</i>	38
Gambar 3.22	<i>Desain Prototipe Android Mobile Menu Bantuan</i>	38
Gambar 4.1	<i>Tampilan Halaman Login</i>	39
Gambar 4.2	<i>Tampilan Halaman Beranda</i>	40
Gambar 4.3	<i>Tampilan Informasi Baru Pada Halaman Beranda</i>	40
Gambar 4.4	<i>Tampilan Informasi Daftar Ulang Tahun Bulan Ini Pada</i>	
Gambar 4.5	<i>Halaman Beranda</i>	41
Gambar 4.6	<i>Tampilan Informasi Jumlah Penjualan Produk Tahun Ini Pada</i>	
	<i>Halaman Beranda</i>	41
Gambar 4.7	<i>Tampilan Grafik Jumlah Penjualan Produk Pertahun Pada</i>	
	<i>Halaman Beranda</i>	41
Gambar 4.8	<i>Tampilan Halaman Daftar Pelanggan</i>	42
Gambar 4.9	<i>Tampilan Halaman Tambah Pelanggan</i>	42
Gambar 4.10	<i>Tampilan Halaman Edit Pelangga</i>	43
Gambar 4.11	<i>Tampilan Halaman Daftar Ulang Tahun</i>	43
Gambar 4.12	<i>Tampilan Halaman Edit Ulang Tahun</i>	44
Gambar 4.13	<i>Tampilan Halaman Daftar Stok Retail</i>	44
Gambar 4.14	<i>Tampilan Halaman Edit Stok Retail</i>	45
Gambar 4.15	<i>Tampilan Halaman Daftar Karyawan</i>	45
Gambar 4.16	<i>Tampilan Halaman Tambah Karyawan</i>	46
Gambar 4.17	<i>Tampilan Halaman Edit Karyawan</i>	46

Gambar 4.18	<i>Tampilan Halaman Daftar Produk</i>	47
Gambar 4.19	<i>Tampilan Halaman Tambah Produk</i>	47
Gambar 4.20	<i>Tampilan Halaman Edit Produk</i>	48
Gambar 4.21	<i>Tampilan Halaman Daftar Pesanan</i>	48
Gambar 4.22	<i>Tampilan Edit Status Pesan</i>	49
Gambar 4.23	<i>Tampilan Daftar Penjualan</i>	49
Gambar 4.24	<i>Tampilan Tambah Data Penjualan</i>	50
Gambar 4.25	<i>Tampilan Edit Data Penjualan</i>	50
Gambar 4.26	<i>Tampilan Daftar Informasi</i>	51
Gambar 4.27	<i>Tampilan Tambah Informasi</i>	51
Gambar 4.28	<i>Tampilan Edit Informasi</i>	52
Gambar 4.29	<i>Tampilan Diagram Jumlah Penjualan Produk Keripik Lele Perbulan</i>	52
Gambar 4.30	<i>Tampilan Statistik Prosentase Penjualan Produk Keripik Lele Perbulan</i>	53
Gambar 4.31	<i>Tampilan Splash Screen</i>	54
Gambar 4.32	<i>Tampilan Form Login</i>	54
Gambar 4.33	<i>Tampilan Form Daftar Pelanggan</i>	55
Gambar 4.34	<i>Tampilan Menu Beranda</i>	55
Gambar 4.35	<i>Tampilan Menu Produk</i>	56
Gambar 4.36	<i>Tampilan Form Pemesanan</i>	56
Gambar 4.37	<i>Tampilan Dialog Detail Pemesanan dan Informasi Pembayaran</i>	57
Gambar 4.38	<i>Tampilan Menu Histori</i>	57
Gambar 4.39	<i>Tampilan Menu Bantuan</i>	58
Gambar 4.40	<i>Tampilan Petunjuk Aplikasi</i>	58
Gambar 4.41	<i>Tampilan Tentang Aplikasi</i>	59
Gambar 4.42	<i>Tampilan Informasi Pembayaran</i>	59
Gambar 4.43	<i>Tampilan Dialog Keluar Aplikasi</i>	60

DAFTAR TABEL

Tabel 4.1 Hasil Pengujian Sistem Pada Aplikasi Web	61
Tabel 4.2 Hasil Pengujian Sistem Pada Aplikasi Mobile	62
Tabel 4.3 Hasil Pengujian Sistem Aplikasi Web dengan Beberapa Web Browser	62
Tabel 4.4 Hasil Pengujian Sistem Aplikasi Mobile Dengan Beberapa Type dan Pixel	64
Tabel 4.5 Hasil Pengujian Sistem Aplikasi Mobile Dengan Beberapa Sistem Operasi	64
Tabel 4.6 Hasil Pengujian Operasi Sales Force Automation Pada Sistem	65
Tabel 4.7 Hasil Pengujian Auto Notification dan Manual Notification Pada Retail	66
Tabel 4.8 Hasil Pengujian Durasi Auto Notification	66

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi dan telekomunikasi akhir-akhir ini telah melahirkan kemungkinan untuk melakukan akses atau pemrosesan informasi secara *mobile* langsung pada lokasi kegiatan operasi. Perkembangan peralatan *mobile device* sangat menunjang efektifitas *user* dalam melakukan kegiatan operasi pada saat dilokasi. *Mobile Device* secara umum harus mampu memenuhi aspek *Ergonomy*, *Reliability*, *Functionality* dan *Connectivity* dengan baik.

Baby Fish adalah sebuah usaha di daerah desa Peniwen, Kecamatan Kromengan, Kabupaten Malang yang bergerak pada pengolahan ikan lele, nila dan wader yang masih berumur dua atau tiga bulan untuk dijadikan keripik. Usaha keripik yang berdiri sejak 14 Maret 2013 ini berkembang begitu pesat sampai saat ini. Untuk mengatasi permasalahan yang timbul pada operasi penjualan keripik *baby fish* secara konvensional, maka *mobile device* seperti *Android* dapat menjadi jawaban yang tepat, dimana untuk penerapannya pada area ini yang secara umum digunakan yaitu *Sales Force Automation (SFA)*.

Sales Force Automation (SFA) adalah suatu sistem yang dapat berdiri sendiri tanpa adanya *Customer Relationship Managemant (CRM)*, sedangkan *Customer Relationship Managemant* hanyalah alat yang membantu suksesnya management *Sales Force Automation* dalam perusahaan. *SFA* termasuk dalam salah satu jenis sistem informasi berbisnis. *SFA* atau Otomasi Tenaga Penjualan berguna sebagai alat untuk mengotomasikan sistem penjualan dan manajemen penjualan.

Android adalah sebuah sistem operasi berbasis *mobile* yang dibangun menggunakan bahasa pemrograman java yang berjalan pada Kernel Linux. Sifatnya yang *mobile* dan merupakan teknologi baru yang mendasari pemilihan teknologi ini. Selain itu, sifatnya yang *open source* yang membuat biaya pengembangan menjadi minimum. Banyak *developer* yang telah mengembangkan sistem informasi yang

mampu berjalan di *android*, maka dari itu penulis mempunyai hipotesis bahwa dapat membuat aplikasi *Sales Force Automation* berbasis *Android*.

1.2 Rumusan Masalah

Adapun rumusan masalah yang dibahas yaitu Bagaimana membangun aplikasi *Sales Force Automation (SFA)* berbasis *Android* agar mampu menunjang efektifitas retail dalam melakukan pembelian keripik *Baby Fish* dan mempermudah *supplier* dalam mengelola informasi yang ada.

1.3 Manfaat

Adapun manfaat yang dapat diperoleh dari penulisan ini adalah sebagai berikut:

1. Efisiensi waktu dalam melakukan pemesanan keripik karena pemesanan dapat dilakukan oleh retail dimanapun berada selama ada koneksi internet.
2. Memberikan kemudahan terhadap *supplier* untuk mengetahui daftar stok keripik pada retail yang masih tersedia.

1.4 Tujuan

Tujuan dari penulisan ini adalah untuk membuat perangkat lunak *Sales Force Automation (SFA)* pada sistem penjualan keripik *Baby Fish* berbasis *Android* untuk menunjang efektifitas retail dalam melakukan pembelian.

1.5 Batasan Masalah

Agar permasalahan yang dibahas terfokus dan tidak melebar, maka diambil batasan masalah sebagai berikut :

1. Bahasa pemrograman yang digunakan dalam membangun aplikasi ini adalah *PHP* dan *MySQL* untuk aplikasi berbasis *web*, sedangkan untuk aplikasi berbasis *mobile* menggunakan bahasa pemrograman *Java*.

2. Teks editor yang digunakan dalam membangun aplikasi *web* menggunakan *Sublime Text 2*, sedangkan pada aplikasi *mobile* menggunakan *Eclipse Indigo*.
3. Pada aplikasi *mobile* hanya dapat melakukan proses pemesanan keripik, melihat daftar promo, melihat info diskon produk dan melihat status pemesanan.
4. Daftar keripik yang tersedia ada tiga jenis yaitu Keripik Lele, Keripik Nila dan Keripik Wader.
5. Ruang lingkup operasi pelayanan pemesanan hanya pada Kabupaten Malang dan Kabupaten Blitar.
6. Studi kasus dilakukan di daerah Desa Peniwen, Kecamatan Kromengan, Kabupaten Malang Pada Usaha Kelompok Tani Kurnia Indah.

1.6 Metodologi Penelitian

Metodologi yang digunakan dalam penulisan tugas akhir ini meliputi dua tahap yaitu sebagai berikut :

1.6.1 Tahap Pengumpulan Data

Metode yang digunakan pada proses pengumpulan data adalah sebagai berikut :

1. Metode Interview

Pengumpulan data dengan melakukan wawancara dan tanya jawab secara langsung kepada pihak terkait dengan objek data penelitian. Metode ini bertujuan untuk memperoleh penjelasan secara langsung tentang data–data yang dipelajari dengan metode pengamatan.

2. Metode Observasi

Pengumpulan data dengan mengadakan pengamatan objek dengan cara melihat secara langsung pada kegiatan yang dilakukan. Metode ini diterapkan dengan tujuan untuk mendapatkan gambaran dan data–data yang jelas dan akurat.

3. Metode Studi Literatur

Pengumpulan data dengan mencari data dari sumber–sumber bacaan seperti : buku, jurnal, referensi, *web page*, *blog*, dan karya tulis ilmiah.

1.6.2 Tahap Pembuatan

Model yang digunakan dalam proses pembuatan aplikasi yaitu dengan menggunakan *waterfall model*. *Waterfall model* adalah salah satu model pengembangan *software*, dimana kemajuan suatu proses dipandang sebagai terus mengalir ke bawah seperti air terjun. Keunggulan model pendekatan pengembangan *software* dengan metode *waterfall* adalah pencerminan kepraktisan rekayasa yang membuat kualitas *software* tetap terjaga karena pengembangannya yang terstruktur dan terawasi.

Disisi lain, model ini merupakan model yang bersifat dokumen lengkap, sehingga proses pemeliharaan dapat dilakukan dengan mudah. Akan tetapi, dikarenakan dokumentasi yang lengkap dan sangat teknis, membuat pihak klien kesulitan membaca dokumen yang berujung pada sulitnya komunikasi antar pengembang dan klien.

Dokumentasi kode program yang lengkap juga secara tak langsung menghapus ketergantungan pengembang terhadap pemrogram yang keluar dari tim pengembang. Hal ini sangat menguntungkan bagi pihak pengembang dikarenakan proses pengembangan perangkat lunak tetap dapat dilanjutkan tanpa bergantung pada *programmer* tertentu. Tahap-tahap pengembangan *waterfall* model adalah sebagai berikut :

1. *System Engineering*

Proses penilaian sistem lama yang sedang berjalan dan studi kelayakan pengembangan sistem baru berdasarkan aspek teknologi, ekonomis dan sumber daya manusia.

2. *Analisis*

Perolehan kebutuhan pengguna sistem dari *user* serta pilihan solusi jenis sistem informasi yang akan dikembangkan.

3. *Desain*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk *software*. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya maka proses ini juga harus di dokumentasikan sebagai konfigurasi dari *software*.

4. *Coding dan Testing*

Desain harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh komputer, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap desain yang secara teknis akan dikerjakan oleh *programmer*. Proses *Coding* ini harus dilakukan *Testing* untuk menguji kesalahan-kesalahan program maupun fungsi dari sistem.

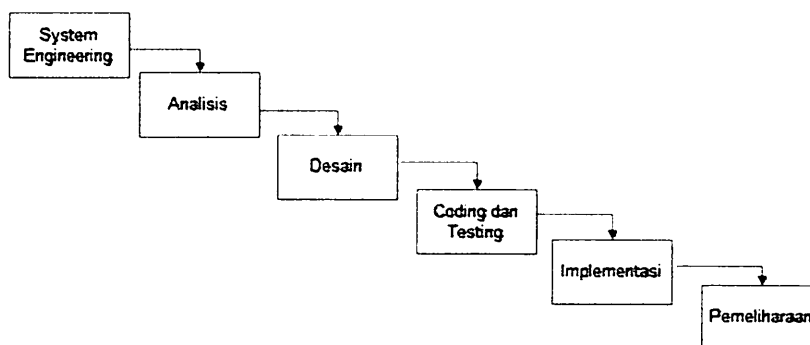
5. *Implementasi*

Setelah semua fungsi-fungsi *software* harus di uji coba agar *software* bebas dari kesalahan, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya. Maka proses selanjutnya adalah bagaimana sistem baru akan diinstall dan dijalankan di perusahaan dengan pengoperasian yang dilakukan oleh *user*.

6. *Pemeliharaan*

Pemeliharaan suatu *software* sangat diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada kesalahan kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari perusahaan seperti ketika ada pergantian sistem operasi atau perangkat lainnya.

Berikut gambaran model pembuatan aplikasi dengan *waterfall* model yang ditunjukkan pada gambar 1.1.



Gambar 1.1 Model Waterfall

Secara keseluruhan model pendekatan pengembangan *software* dengan metode *waterfall* cocok untuk pengembangan *software* / perangkat lunak dengan resiko yang kecil dan memiliki ukuran yang kecil serta waktu pengembangan yang cukup panjang. Model ini tidak disarankan untuk ukuran perangkat lunak yang besar dan tingkat resiko yang besar.

1.7 Sistematika Penulisan

Sistematika penyusunan laporan skripsi ini untuk memberi gambaran secara umum dari laporan skripsi yang dibuat meliputi bab–bab sebagai berikut :

BAB I : PENDAHULUAN

Pada bab ini memuat Latar Belakang, Rumusan Masalah, Tujuan, Manfaat, Batasan Masalah, Metodologi Penelitian dan Sistematika Penulisan.

BAB II : LANDASAN TEORI

Pada bab ini akan dibahas tentang sistem informasi yang digunakan, database yang digunakan untuk menyimpan data serta software pendukung yang akan digunakan untuk pengembangan sistem yang dapat dijadikan untuk pemecahan masalah.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan membahas tentang tinjauan umum, desain sistem, perancangan sistem, hasil perancangan, analisis kelayakan, analisis sistem dan evaluasi.

BAB IV : IMPLEMENTASI DAN PEMBAHASAN

Pada bab ini akan membahas tentang hasil implementasi dan analisis hasil uji coba program. Memaparkan hasil–hasil dari tahap penelitian, tahap analisis, desain, implementasi desain, hasil *testing* dan implementasinya berupa penjelasan teoritik, baik

secara kualitatif, kuantitatif atau secara statistik. Selain itu membandingkan dengan hasil penelitian yang masih manual.

BAB V : PENUTUP

Pada bab ini akan membahas tentang kesimpulan dan saran. Kesimpulan diambil dari data–data hasil pengujian, apakah hasil yang didapat berhasil mengembangkan sistem yang sudah ada dan apakah hasil pengembangan sudah layak untuk di implementasikan.

BAB II

LANDASAN TEORI

2.1 Baby Fish

Baby Fish adalah sebuah usaha di daerah desa Peniwen, Kecamatan Kromengan, Kabupaten Malang yang bergerak pada pengolahan ikan lele, nila dan wader yang masih berumur dua atau tiga bulan untuk dijadikan keripik. Usaha keripik yang berdiri sejak 14 Maret 2013 ini berkembang begitu pesat sampai saat ini. Sebuah usaha yang masih tergolong baru dan banyak diminati oleh pelanggan dan retail-retail di seluruh Kota Malang, khususnya sebuah rumah makan.

Usaha ini pertama kali dikembangkan oleh seorang pengusaha bernama Bapak Sutrio bersama istrinya. Setiap harinya, usaha ini mampu memproduksi 20 kg keripik *Baby Fish* yang sebagian besar adalah keripik lele. Proses produksi mulai dari memanen ikan yang sudah berumur satu bulan, kemudian di olah dan digoreng dengan tepung terigu. Setelah digoreng, dilakukan proses pengeringan dengan menggunakan alat pengering yaitu *spring* untuk menghilangkan minyak-minyak yang terdapat pada keripik agar keripik dapat bertahan lama dalam kemasan.

Banyaknya permintaan dari pelanggan membuat pengusaha *Baby Fish* menjadi kualahan dikarenakan produksi yang semakin meningkat dengan kapasitas kolam yang terbatas dan jumlah ikan yang masih minim, membuat suatu ide untuk menerapkan sistem panggung, dimana setiap rumah pada desa Peniwen ini dibangun kolam ikan, dan pada saat ikan sudah berumur dua atau tiga bulan, pengusaha *Baby Fish* menebas ikan-ikan tersebut, sehingga kapasitas ikan yang tersedia dapat memenuhi produksi sesuai permintaan pelanggan.

Pada saat ini pengembangan usaha keripik *Baby Fish* dikelola oleh beberapa kelompok tani di desa Peniwen yang dikenal dengan istilah Kelompok Tani Kurnia Indah. Tujuan dibentuknya kelompok tani untuk lebih meningkatkan produksi keripik agar mampu memenuhi permintaan yang semakin banyak. Selain itu, juga untuk merubah dan memajukan pola pikir masyarakat untuk mampu bersaing di dunia bisnis

pada zaman era globalisasi saat ini agar tercapai tujuan bersama menjadikan desa Peniwen sebagai *central* bisnis atau yang dikenal sebagai *home industri*.

2.1.1 Visi dan Misi

Visi dari Usaha Keripik *Baby Fish* adalah untuk mengembangkan usaha bisnis yang bergerak di bidang pengolahan keripik ikan agar mampu meningkatkan produksi dan menjadikan desa Peniwen sebagai *central* bisnis atau *home industri*.

Misi dari Usaha Keripik *Baby Fish* adalah memberikan pelayanan produksi bagi para pelanggan dan retail-retail pada rumah makan dengan sebaik-baiknya agar hubungan bisnis selalu terjaga dan semakin meningkat.

2.1.2 Logo Baby Fish

Logo merupakan suatu bentuk gambar atau sekedar sketsa dengan arti tertentu, dan mewakili suatu arti dari perusahaan, daerah, perkumpulan, produk, negara, lembaga atau organisasi dan hal-hal lainnya yang dianggap membutuhkan hal yang singkat dan mudah diingat sebagai pengganti dari nama sebenarnya.

Logo juga merupakan suatu media yang sangat penting yang harus dimiliki oleh suatu badan usaha, karena logo tersebut digunakan sebagai media untuk pengenalan merk dan identitas suatu usaha. Logo akan menjadi semakin diperlukan disaat promo dijalankan. Berikut adalah logo dari Usaha Keripik *Baby Fish* yang ditunjukkan pada gambar 2.1.



Gambar 2.1 Logo Usaha Keripik *Baby Fish*

2.2 Sales Force Automation

Sales Force Automation (SFA) merupakan jenis program yang mengotomasi pekerjaan atau tugas-tugas bisnis seperti mengontrol inventori, proses penjualan, pelacakan interaksi pelanggan dan menganalisa proyeksi penjualan beserta kinerjanya. Akan tetapi sekarang banyak perusahaan menerapkan *SFA* sebagai sebuah aplikasi penjualan barang atau media retail pemesanan barang terhadap *supplier*. *SFA* dikembangkan sesuai kebutuhan bisnis yang ada pada sebuah *corporate*. *SFA* sendiri sering disebut sebagai *Sales Automation Software* atau *CRM Software*.^[6]

Sales Force Automation (SFA), yang mulai tersedia pada pertengahan tahun 80-an adalah komponen pertama dari *CRM*. *SFA* membantu para tenaga penjualan untuk mengatur daftar *account* dan *contact* serta mengatur *track opportunities* mereka. *SFA* juga dapat digunakan untuk mengawasi alur penjualan yang ada dan membantu mengoptimalkan penyampaian informasi dengan adanya pembagian informasi antar tenaga penjual.^[7]

SFA adalah sistem yang secara otomatis mencatat semua tahap dalam proses penjualan. Sistem manajemen kontak mampu melacak semua kontak yang telah dibuat sesuai pelanggan yang ada, tujuan dari kontak, dan segala tindak lanjut yang mungkin diperlukan. Hal ini memastikan bahwa usaha penjualan tidak diduplikasi, mengurangi risiko yang membuat kesal pelanggan. Elemen lain dari sistem *SFA* dapat mencakup peramalan penjualan (*sales forecasting*), manajemen pemesanan dan pengetahuan produk.^[7]

Fungsi besar dari *SFA* adalah untuk memaksimalkan penjualan dan fokus layanan terhadap pelanggan. Pelanggan dimungkinkan untuk melakukan pemesanan barang melalui sebuah sistem yang berbasis *mobile*. Komponen teknologi *SFA* adalah jaringan, *hardware* dan *software*. Proses tersebut dapat dipermudah dengan menggunakan aplikasi *SFA mobile*, sehingga aplikasi tersebut memungkinkan retail-retail untuk melakukan pemesanan barang melalui handphone *Android* mereka.^[6]

SFA mengelola berbagai aktivitas penjualan perusahaan. Proses-proses penjualan dapat diuraikan menjadi beberapa tahapan, misalnya menjaring prospek (*lead generation*), kualifikasi prospek (*lead qualification*), pembuatan proposal, dan realisasi penjualan. Proses-proses ini pada *SFA* dirangkul menjadi *opportunity management*,

contact manajement, pembuatan proposal, dan konfigurasi produk. *Opportunity mangement* biasanya memiliki fitur *lead management* (manajemen prospek) dan *sales forecasting* (estimasi skala penjualan di masa mendatang).^[8]

2.3 Aplikasi *Mobile* ^[5]

Sistem aplikasi *mobile* merupakan aplikasi yang dapat digunakan walaupun pengguna berpindah dengan mudah dari satu tempat ke tempat lain tanpa terjadi pemutusan atau terputusnya komunikasi. Aplikasi ini dapat diakses melalui perangkat nirkabel seperti *pager*, telepon seluler dan *PDA (Personal Digital Assistant)*.

Mayoritas aplikasi *mobile* menggunakan *Java Micro Edition (JME) platform*, yang dikembangkan untuk perangkat kecil seperti ponsel dan *set-top box*. *Java Micro Edition (JME)* diperkecil menggunakan *subset* dari *Java SE* komponen, *virtual machine* dan *API*. Ini juga mendefinisikan *API* yang ditargetkan pada konsumen khusus *mobile* dan perangkat *embedded*. *NetBeans Java Micro Edition (JME)* mendukung dua konfigurasi dasar dari *platform Java Micro Edition (JME)*, *Connected Limited Device Configuration (CLDC)* dan *CDC*.

Adapun karakteristik perangkat *mobile* yaitu :

1. Ukuran yang kecil : Perangkat *mobile* memiliki ukuran yang kecil. Konsumen menginginkan perangkat yang terkecil untuk kenyamanan dan mobilitas mereka.
2. Memory yang terbatas : Perangkat *mobile* juga memiliki *memory* yang kecil, yaitu *Primary (RAM)* dan *Secondary (Disk)*.
3. Daya proses yang terbatas : Sistem *mobile* tidaklah setangguh rekan mereka yaitu *desktop*.
4. Mengonsumsi daya yang rendah : Perangkat *mobile* menghabiskan sedikit daya dibandingkan dengan mesin *desktop*.
5. Kuat dan dapat diandalkan : Karena perangkat *mobile* selalu dibawa kemana saja, mereka harus cukup kuat untuk menghadapi benturan-benturan, gerakan, dan sesekali tetesan-tetesan air.
6. Konektivitas yang terbatas : Perangkat *mobile* memiliki *bandwith* rendah, beberapa dari mereka bahkan tidak tersambung.

2.4 Android ^[3]

Android adalah sebuah sistem operasi pada handphone yang bersifat terbuka dan berbasis pada sistem operasi *Linux*. *Android* bisa digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *TMobile*, dan *Nvidia*. Pada saat perilisan perdana *Android*, 5 November 2007, 10 *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode *android* di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Android dipuji sebagai “*platform mobile* pertama yang lengkap, terbuka dan bebas” karena *android* merupakan generasi baru dari *platform mobile*, yaitu *platform* yang memberikan pengembang untuk melakukan pengembangan sesuai dengan apa yang diharapkan. Sistem operasi yang mendasari *android* dilisensikan dibawah *GNU, GPLv2 (General Public Lisensi Versi 2)*, yang sering dikenal dengan istilah “*Copyleft*”. Sebagian pengembang menggunakan *Eclipse* yang secara bebas untuk merancang dan mengembangkan aplikasi *android*.

Sistem operasi *android* dirilis sebagai berikut:

1. *Android* versi 1.1 (*Bender*) dirilis pada tanggal 9 Maret 2009 oleh *google* yang dilengkapi dengan pembaruan jam *alarm* dan *voice search*.
2. *Android* versi 1.5 (*Cupcake*) dirilis pertengahan mei 2009 dengan banyak pembaruan diantaranya kemampuan merekam dan menonton video dengan modus kamera, dukungan *Bluetooth A2DP* dan yang lainnya.
3. *Android* versi 1.6 (*Donut*) dirilis September 2009 dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan *control applet* VPN dan resolusi WVGA.

4. *Android* versi 2.0/2.1 (*Enclair*) diluncurkan pada 3 Desember 2009 dilakukan perubahan yaitu pengoptimalan *hardware*, perubahan *user interface* (UI) dengan *browser* baru dan dukungan HTML5.
5. *Android* versi 2.2 (*Froyo*) pada tanggal 20 Mei 2010 kembali diluncurkan ponsel *android* versi 2.2 (*Froyo*) perubahan yang dilakukan meliputi optimasi kecepatan, memori, dan kinerja sistem operasi secara keseluruhan.
6. *Android* versi 2.3 (*Gingerbread*) 1 Desember 2010 *Google* kembali meluncurkan versi terbaru yaitu *Android* versi 2.3. Pada versi ini peningkatan daya, kontrol melalui aplikasi, penggunaan *multiple* kamera, peningkatan performa dan penambahan sensor seperti *gyroscope*.
7. *Android* versi 3.0/3.1 (*Honeycomb*) versi ini berbeda dengan versi-versi sebelumnya. Versi ini dirancang khusus untuk PC Tablet sehingga memiliki *user interface* yang berbeda dan mendukung ukuran layar yang lebih besar. Selain itu, pada versi ini memungkinkan penggunaan *multiprosesor* dan akselerasi perangkat keras untuk grafis, versi berikutnya yang muncul yaitu versi Ice Cream pada kuartal keempat 2011.
8. *Android* versi 4.0 (*Ice Cream Sandwich*) Pertama kali dirilis pada 19 Oktober 2011. *Android* pertama yang mempunyai fitur baru membuka kunci dengan pengenalan wajah. Fitur ini belum dimiliki oleh *android* versi dibawah ICS tetapi sudah disempurnakan pada versi di atasnya. ICS memiliki tampilan *Interface* yang lebih bersih dan *smooth*. Penambahan *font* yang bernama *Roboto* yang mana *font* ini sangat simpel dan elegan apabila digunakan dan terkesan sangat unik.
9. *Android* versi 4.1 - 4.3 (*Jelly Bean*) dirilis pada tanggal 24 Juli 2013. Memiliki performa yang lebih cepat, *widget* yang bisa di *resize*, mampu mengetik dengan suara, mencari dengan suara, notifikasi yang lebih kaya, memiliki *google now*, *face unlock* dengan *Liveness Check*, teknologi *Project Butter* untuk meningkatkan responsifitas yang sangat baik, sistem operasi yang cepat dan ringan dan tampilan *google play* yang lebih segar.
10. *Android* versi 4.4 (*Kitkat*) dirilis pada tanggal 28 Oktober 2013. Desain antarmuka transparan dengan fitur untuk ngeprint yang dapat dikoneksikan dengan fitur *Google Cloud Print* untuk mencetak dokumen langsung dari *Smartphone*, mode

pada kamera lebih canggih dan *hangouts* lebih terintegrasi. Untuk fitur *voice search* performa lebih cepat dan penggunaan *google now* lebih akurat.

2.5 PHP (*PHP Hypertext Preprocessor*)

PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya adalah yang dikirimkan ke klien, tempat pemakai menggunakan *browser*.^[2]

Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, dapat membentuk suatu tampilan berdasarkan permintaan teknisi. Misalnya, kita bisa menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, dan *Perl*. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *command line*. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*.^[2]

PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. jika dilihat dari versi pertamanya bahwa PHP terdiri dari sekumpulan *script PERL* yang digunakan untuk mengolah data *form* dari *web*. Kemungkinan bahwa PHP singkatan dari *Perl Hypertext Preprocessor*. Pada awalnya PHP bernama FI (*Form Interpreted*). Setelah Rasmus melepaskan kode sumbernya, maka terbentuklah nama PHP/FI (*Personal Home Page / Form Interpreter*). Sejak itulah PHP bersifat *open source*.^[10]

Pada saat ini PHP cukup populer sebagai piranti pemrograman *web*, terutama di lingkungan *Linux*. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada *server-server* yang berbasis *UNIX*, *Windows*, dan *Macintosh*.^[2]

Pada awalnya PHP dirancang untuk diintegrasikan dengan *web server Apache*. Namun, belakangan PHP juga dapat bekerja dengan *web server* seperti PWS (*Personal Web Server*), IIS (*Internet Information Server*), dan *Xitami*.^[2]

pada tahun lebih cepat dan biaya lebih rendah. Untuk itu perlu
sangat penting lebih cepat dan terjangkau biaya untuk

2.1.1.1. (WIP: WIP Processor)

WIP merupakan singkatan dari WIP (WIP Processor) dan WIP merupakan
kegiatan yang dilakukan dalam proses di server yang
dapat dikurir ke klien untuk pemrosesan.

Secara khusus WIP dirancang untuk membentuk aplikasi web dan
dapat membentuk server aplikasi berdasarkan platform berbasis
memiliki dan biaya yang rendah. WIP merupakan modul
yang sangat penting seperti ASP (Active Server Page) dan ASP
dalam proses pembuatan WIP sehingga bisa di pakai secara langsung
tanpa perlu WIP dapat diinstal dan pemrosesan web server yang
dapat.

WIP pertama kali dibuat oleh James F. Wood pada tahun 1995. WIP
versi pertama yang WIP terdiri dari sekumpulan server WWW yang
maka mengolah data yang dari web. Keunggulannya WIP singkatan dari WIP
WIP Processor. Pada awalnya WIP bernama WIP (WIP Processor) dan
kemudian kode sumbernya maka terbentuk nama WIP (WIP Processor).

WIP (WIP Processor) sejak itulah WIP (WIP Processor) dan WIP
pada saat ini WIP cukup populer sebagai platform pemrosesan web. WIP
keunggulan WIP. Walaupun demikian WIP sebenarnya juga dapat di pakai
keunggulan yang basis WWW, WIP dan WIP.

Pada tahun WIP dirancang untuk diinstal dan di pakai secara
kemudian WIP juga dapat bekerja dengan web server seperti WWW (WIP
WIP (WIP Processor) dan WIP.

2.6 MySQL

MySQL adalah sebuah program *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi *user*, serta menggunakan perintah standar *SQL (Structured Query Language)*.^[9]

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama *MySQL AB* yang pada saat itu bernama *TcX DataKonsult AB* sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak 1797. Awalnya *TcX* membuat *MySQL* dengan tujuan mengembangkan aplikasi *web* untuk klien. *TcX* merupakan perusahaan pengembang *software* dan konsultan *database*. Saat ini *MySQL* sudah diakuisisi oleh *Oracle Corp.*^[1]

MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya. Kepopuleran *MySQL* antara lain karena *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database*-nya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan skala menengah-kecil. *MySQL* juga bersifat *open source* dan *free* (Anda tidak perlu membayar untuk menggunakannya) pada berbagai *platform* (kecuali pada *Windows*, yang bersifat *Shareware*). *MySQL* didistribusikan dengan lisensi *open source GPL (General Public License)* mulai versi 3.23, pada bulan Juni 2000.^[1]

MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (*PHP* dan *Perl*). *MySQL* dan *PHP* dianggap sebagai pasangan *software* pengembangan aplikasi *web* yang ideal. *MySQL* lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script PHP*.^[1]

2.7 XAMPP^[4]

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan

web server yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

XAMPP adalah singkatan yang masing-masing hurufnya adalah :

1. *X* yang berarti Program ini dapat dijalankan di banyak sistem operasi seperti *Windows, Linux, Mac OS, dan Solaris*.
2. *A* yang berarti *Apache*, yang merupakan aplikasi *web server*. Tugas utama *Apache* adalah menghasilkan halaman *web* yang benar kepada *user* berdasarkan kode PHP yang dituliskan oleh pembuat halaman *web*. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu *database* diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman *web* yang dihasilkan.
3. *M* yang berarti MySQL, aplikasi *database server*. Perkembangannya disebut SQL yang merupakan kepanjangan dari *Structured Query Language*. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah *database*. MySQL dapat digunakan untuk membuat dan mengelola *database* beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah, dan menghapus data yang berada dalam *database*.
4. *P* yang berarti PHP, bahasa pemrograman *web*. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *web* yang bersifat *server IDE scripting*. PHP memungkinkan kita untuk membuat halaman *web* yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP adalah MySQL. Namun PHP juga mendukung sistem manajemen *database* Oracle, Microsoft Access, Interbase, dBase, PostgreSQL, dan sebagainya.
5. *P* yang berarti Perl, yaitu suatu bahasa pemrograman.

2.8 Eclipse ^[3]

Eclipse merupakan *IDE (Integrated Development Environment)* untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform - independent*).

Berikut ini sifat dari *Eclipse* :

1. *Multi – Platform* : target sistem operasi *Eclipse* adalah *Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X*.

2. *Multi Language* : *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti *C/C++*, *Cobol*, *Python*, *Perl*, *PHP* dan lain sebagainya.
3. *Multi-Role* : *Eclipse* dapat digunakan untuk aktifitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, *test* perangkat lunak, pengembangan *web* dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*. *Eclipse* dibuat dari kerja sama antara perusahaan-perusahaan anggota '*Eclipse Foundation*' (berserta individu-individu lain). Banyak nama besar yang ikut dalam '*Eclipse Foundation*', termasuk *IBM*, *BEA*, *Intel*, *Nokia*, *Borland*. *Eclipse* bersaing langsung dengan *Netbeans IDE*. *Plugin* tambahan pada *Eclipse* jauh lebih banyak dan bervariasi dibandingkan IDE lainnya.

Eclipse awalnya dikembangkan oleh *IBM* untuk menggantikan perangkat lunak *IBM Visual Age for Java 4.0*. Produk ini diluncurkan oleh *IBM* pada tanggal 5 November 2001, yang menginvestasikan sebanyak *US\$ 40 juta* untuk pengembangannya. Semenjak itu konsorsium *Eclipse Foundation* mengambil alih untuk pengembangan *Eclipse* lebih lanjut dan pengaturan organisasinya. Sejak versi 3.0, *Eclipse* pada dasarnya merupakan sebuah kernel, yang mengangkat *plug-in*. Apa yang dapat digunakan di dalam *Eclipse* sebenarnya adalah fungsi dari *plug-in* yang sudah diinstal. Ini merupakan basis dari *Eclipse* yang dinamakan *Rich Client Platform (RCP)*. Berikut ini adalah komponen yang membentuk *RCP*:

1. *Core platform*
2. *OSGi*
3. *SWT (Standard Widget Toolkit)*
4. *Jface*
5. *Eclipse Workbench*

2.9 Java ^[3]

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun *Microsystems* yang saat ini merupakan bagian dari *Oracle* dan dirilis pada tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis *Java* umumnya dikompilasi ke dalam *pcode* (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM).

Java merupakan bahasa pemrograman yang bersifat umum/non - spesifik (*general purpose*), dan secara khusus didesain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi *Java* mampu berjalan di beberapa *platform* sistem operasi yang berbeda, *Java* dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini *Java* merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis *web*.

Bahasa pemrograman *Java* terlahir dari *The Green Project*, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan *Oak*. Proyek ini dimotori oleh Patrick Naughton, Mike Sheridan, dan James Gosling, beserta sembilan pemrogram lainnya dari *Sun Microsystems*. Salah satu hasil proyek ini adalah maskot Duke yang dibuat oleh Joe Palrang.

Pertemuan proyek berlangsung di sebuah gedung perkantoran Sand Hill Road di Menlo Park. Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program *Java Oak* pertama, yang ditujukan sebagai pengendali sebuah peralatan dengan teknologi layar sentuh (*touch screen*), seperti pada PDA sekarang ini. Teknologi baru ini dinamai dengan "*7" (*Star Seven*). Setelah era *Star Seven* selesai, sebuah anak perusahaan TV kabel tertarik ditambah beberapa orang dari proyek *The Green Project*. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, Palo Alto.

Perusahaan baru ini bertambah maju dengan jumlah karyawan yang meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada rentang waktu ini juga ditetapkan pemakaian Internet sebagai medium yang menjembatani kerja dan ide di antara mereka. Pada awal tahun 1990-an, Internet masih merupakan rintisan, yang dipakai hanya di kalangan akademisi dan militer. Mereka menjadikan peramba (*browser*) *Mosaic* sebagai landasan awal untuk membuat perambah *Java* pertama yang dinamai *Web Runner*, terinspirasi dari film 1980-an, *Blade Runner*. Pada perkembangan rilis pertama, *Web Runner* berganti nama menjadi *Hot Java*.

Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber *Java* versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar *San Jose Mercury News* pada tanggal 23 Mei 1995. Sayangnya terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah ruangan hotel Sheraton Palace. Tiga dari pimpinan utama proyek, Eric Schmidt dan George Paolini dari *Sun Microsystems* bersama Marc Andreessen, membentuk *Netscape*. Nama *Oak*, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja “Bapak Java”, James Gosling. Nama *Oak* ini tidak dipakai untuk versi *release Java* karena sebuah perangkat lunak lain sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi “*Java*”. Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling. Konon kopi ini berasal dari Pulau Jawa. Jadi nama bahasa pemrograman *Java* tidak lain berasal dari kata Jawa (bahasa Inggris untuk Jawa adalah *Java*).

2.10 JDK (*Java Development Kit*)^[3]

Java Development Kit (JDK) merupakan perangkat lunak yang digunakan untuk manajemen dan membangun berbagai aplikasi *Java*. *JDK* merupakan superset dari *Java Runtime Environment (JRE)*, berisikan segala sesuatu yang ada di *JRE* ditambahkan compiler dan debugger yang diperlukan untuk mengembangkan applet dan aplikasi.

2.11 SDK (*Software Development Kit*)^[3]

Standart Development Kit (SDK) adalah kumpulan dari tools yang dibutuhkan untuk membuat serta menjalankan program. Jadi di dalam *SDK* terdapat *Java*

Perencanaan baru ini bertentangan dengan jumlah lapangan yang meningkat dalam waktu singkat dan 13 provinsi. Pada rencana yang sedang ditinjau, pembangunan tersebut sangat penting yang menunjukkan bahwa ada di antara mereka. Pada awal tahun 1990-an, faktor-faktor masih merupakan masalah yang dihadapi bangsa di kalangan akademisi dan militer. Mereka menyalahkan pemerintah (Wahono, 1990). Langkah awal untuk membangun kembali Jawa pertama yang dituangkan dalam rencana ini ditinjau dari tahun 1980-an. Wakil Gubernur Jawa Barat memperhatikan nilai potensial Jawa Barat sebagai bagian dari Jawa Barat.

Pada sekitar tahun 1995 untuk pertama kali kode sumber Jawa Barat 1995 diterbitkan. Suksesnya mereka dibantu dengan untuk pembangunan pertama kali pada saat tahun Jawa Barat sebagai Jawa Barat tanggal 22 Mei 1995. Rancangan tersebut merupakan di antara mereka untuk hari pada tahun di sebuah kawasan hotel, restoran, toko, Tiga dari pimpinan utama proyek, yaitu Soeharto dan George Pabedjo dan Soeharto. Masyarakat di antara lain: Andriyanto, mentransfer ke Jawa Barat dan diambil dari lokasi yang termasuk di dalam rencana manajemen kerja "Jawa Barat James Gosling". Nama ini tidak dipakai untuk versi awal karena sebagian orang merasa bahwa ini adalah terdistribusikan merek dagang tersebut sehingga di antara orang-orang yang terlibat. Nama ini diambil dari kota tempat yang digiling yang sangat dan biji-bijian (dari biji-bijian) kesukaan Gosling. Karena kopi ini berasal dari pulau Jawa, jadi nama bahasa program nama Jawa tidak lain berasal dari kata Jawa (bahasa Inggris merek Jawa adalah Jawa).

2.10. JRE (Java Development Kit)

Java Development Kit (JDK) merupakan perangkat lunak yang digunakan untuk membangun dan menjalankan aplikasi. Java JDK merupakan susunan dari kelas-kelas, Antarmuka (API), dan kelas-kelas lainnya yang ada di dalamnya. compiler dan debugger yang diperlukan untuk mengembangkan program dan aplikasinya.

2.11. SDK (Software Development Kit)

Software Development Kit (SDK) adalah kumpulan dari tools yang dibutuhkan untuk membuat serta menjalankan program. Hal ini dalam SDK termasuk Java

Development Kit (JDK) dan *Java Runtime Environment (JRE)*, serta mungkin ada *Integrated Development Environmnet (IDE)* nya juga. Beberapa IDE yang biasa dipakai untuk *JAVA* antara lain *Eclipse*, *Netbeans*, *JBuilder*, dll.

2.12 AVD (*Android Virtual Device*)^[3]

Android Virtual Devices (AVD) adalah konfigurasi dari *emulator* sehingga kita dapat menjalankan perangkat *Android* sesuai model yang dipilih. Dimana setiap AVD terdiri dari :

1. Sebuah profil perangkat keras yang dapat mengatur opsi untuk menentukan fitur *hardware emulator*. Misal, kita dapat menentukan apakah menggunakan perangkat kamera, apakah menggunakan *keyboard* QWERTY fisik atau tidak, berapa banyak memori internal, dan lain-lain.
2. Sebuah pemetaan versi *Android* dapat menentukan versi dari *platform Android* yang akan berjalan pada *emulator*.
3. Pilihan lainnya kita dapat menentukan skin yang ingin digunakan pada *emulator*, yang memungkinkan untuk menentukan dimensi layar, tampilan, dan sebagainya. Dapat menentukan *SD Card* virtual untuk digunakan pada *emulator*.

BAB III

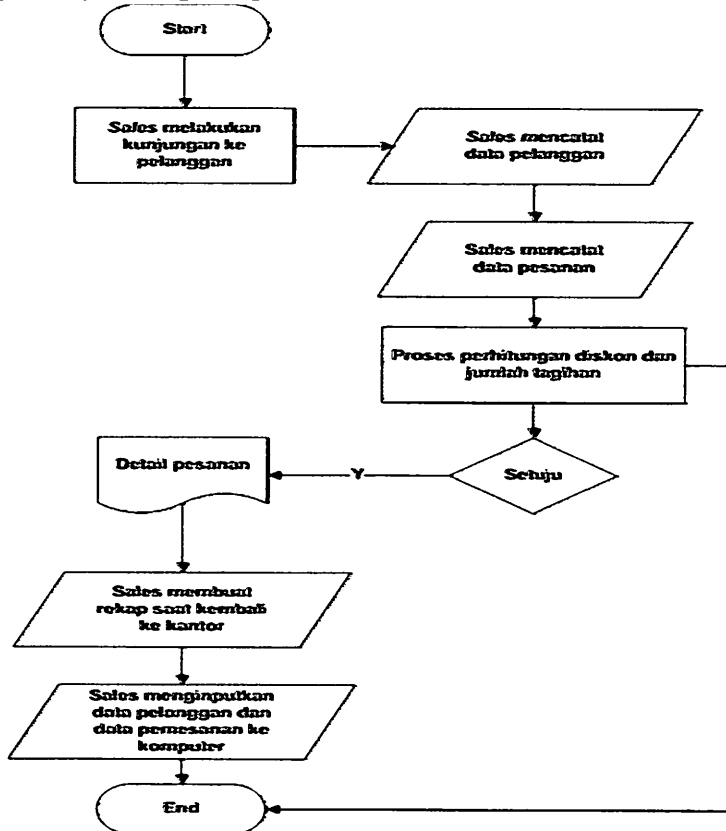
ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

Pada analisis sistem dilakukan analisa operasi penjualan dan analisa kebutuhan seperti analisis operasi penjualan konvensional, analisis operasi *sales force automation*, analisis kebutuhan data dan analisis kebutuhan perangkat lunak.

3.1.1 Analisis Operasi Penjualan Konvensional

Pemakaian aplikasi komputer sudah umum di berbagai kalangan pelaku industri. Namun pada usaha *Baby Fish* masih bersifat *office-centric* dimana pemasukkan data penjualan dilakukan secara manual. Akibatnya *sales* harus melakukan pencatatan dan pembuatan dokumen (tanda terima, faktur, bukti order dan lain-lain) sesuai transaksi yang terjadi. Berikut gambaran secara ringkas proses operasi penjualan secara konvensional yang ditunjukkan pada gambar 3.1.



Gambar 3.1 Flowchart Operasi Penjualan Konvensional

Data yang ditulis secara manual tersebut juga masih perlu dibuatkan rekapnya secara manual, sebagai bagian dari pelaporan hasil kegiatan penjualan hari itu. Selanjutnya dokumen manual yang diserahkan akan dimasukkan ke komputer yang umumnya baru dilakukan keesokan hari. Mekanisme operasi penjualan seperti ini menimbulkan banyak masalah, seperti:

1. Waktu *processing* yang lama di tempat pelanggan dan sangat mengurangi produktifitas penjualan.
2. Harus melakukan kalkulasi *discount*, penjumlahan total *amount* dan lain-lain secara manual.
3. Waktu yang cukup lama untuk pembuatan rekap saat kembali ke kantor.
4. Tingkat kesalahan yang relative tinggi, karena bentuk tulisan tangan yang tidak selalu jelas, padahal data pada dokumen tersebut akan dibaca dan di-*entry* ke komputer oleh orang *sales*.

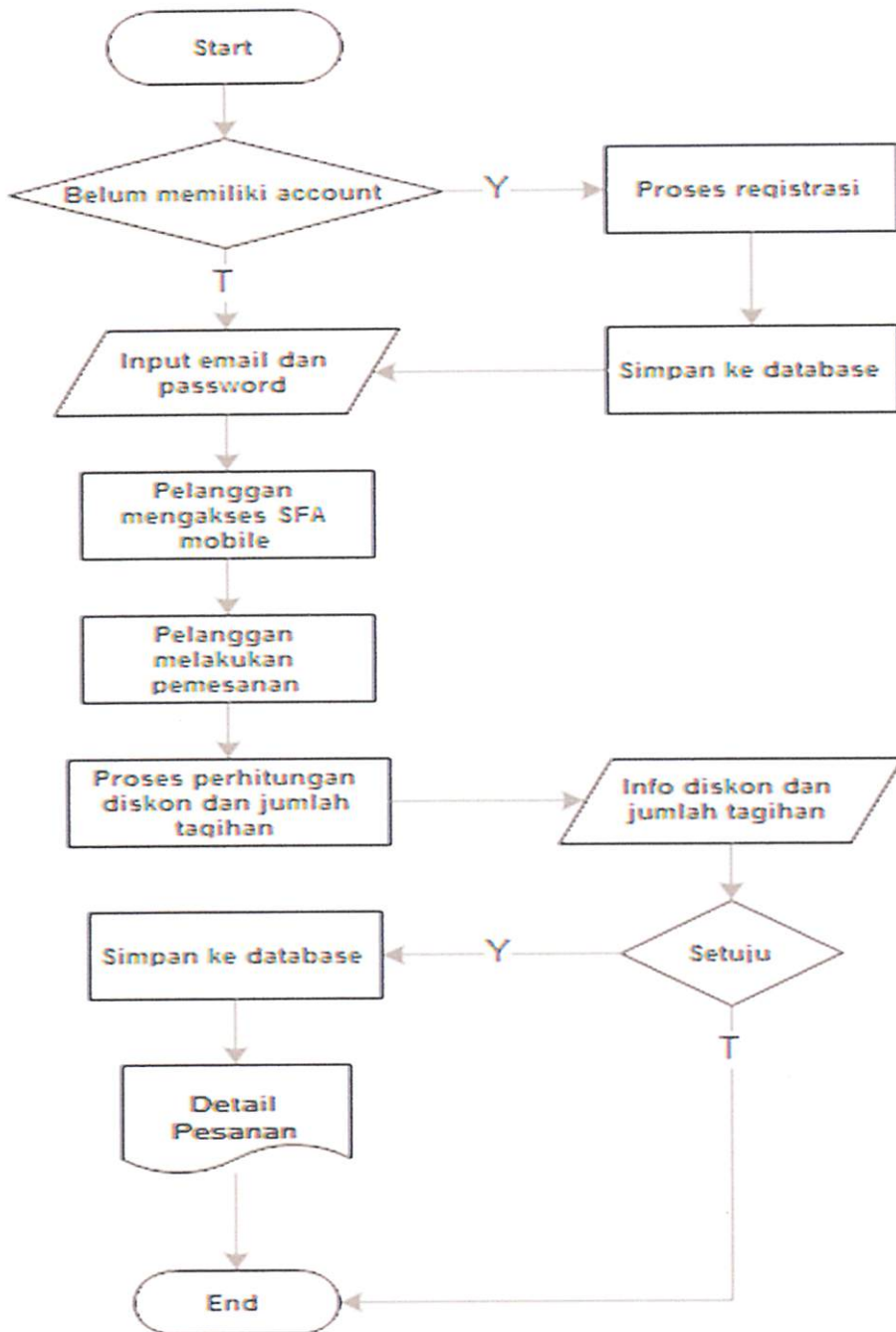
3.1.2 Analisis Operasi Sales Force Automation

Dalam penerapan *Sales Force Automation*, pelanggan dapat melakukan pemesanan langsung secara *mobile*. Sinkronisasi data dengan aplikasi dapat dilakukan langsung di tempat pelanggan.

Berikut prosedur-prosedur yang dilakukan dalam pemesanan keripik secara *mobile*:

1. Pelanggan login terlebih dahulu pada aplikasi *mobile*. Apabila belum memiliki *account*, pelanggan harus melakukan registrasi data.
2. Pelanggan melakukan pemesanan keripik dengan memilih jenis keripik. Pemasukan data pemesanan cukup dengan memasukkan nama produk dan *quantity*.
3. Aplikasi *Sales Force Automation* secara otomatis memberikan info perhitungan diskon dan jumlah tagihan.

Berikut gambaran secara ringkas proses operasi *Sales Force Automation* yang ditunjukkan pada gambar 3.2.



Gambar 3.2 Flowchart Operasi Sales Force Automation

Mekanisme operasi penjualan seperti yang ditunjukkan pada gambar 3.2, memiliki beberapa kelebihan seperti waktu *processing* yang jauh lebih singkat, karena pemesanan dapat dilakukan langsung ditempat pelanggan, tingkat akurasi data yang tinggi, sehingga data yang masuk dapat diyakini kebenarannya dan secara tidak langsung juga dapat mendukung peningkatan kepuasan pelanggan.

3.1.3 Analisis Kebutuhan Data

Sistem yang dibangun adalah aplikasi *Sales Force Automation* pemesanan keripik. Aplikasi *mobile* digunakan untuk melakukan pemesanan keripik, maka *server* harus menyediakan daftar data produk agar aplikasi dapat mengakses data sesuai pesanan yang di inginkan pelanggan.

3.1.4 Analisis Kebutuhan Perangkat Lunak

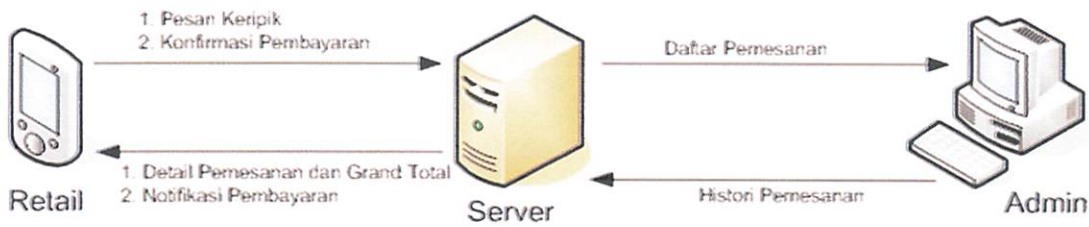
Perangkat lunak yang digunakan untuk membangun aplikasi *Sales Force Automation* pemesanan keripik ini adalah sebagai berikut :

1. *Windows 7 Home Premium 32 Bit* sebagai sistem operasi.
2. *XAMPP*, sebagai *web server* secara *offline*.
3. *IDE Eclipse Indigo*, perangkat lunak yang digunakan untuk membangun aplikasi *mobile*.
4. *Sublime Text 2*, perangkat lunak yang digunakan untuk membangun aplikasi *server* berbasis *web*.
5. *Adobe Photoshop 7*, perangkat lunak yang digunakan untuk membuat logo, gambar, *header*, *footer*, *line*, *tab* dan lain - lain untuk desain perangkat lunak.
6. *Java Development Kit (JDK)*, perangkat lunak yang digunakan untuk kompilasi kode-kode program.
7. *Android Software Development Kit (SDK)*, perangkat lunak yang digunakan untuk pengembang aplikasi android.
8. *Android Virtual Device (AVD)*, perangkat lunak yang digunakan sebagai emulator.
9. *Mozilla Firefox*, sebagai *web browser* untuk menjalankan aplikasi *server* berbasis *web*.

3.2 Perancangan Sistem

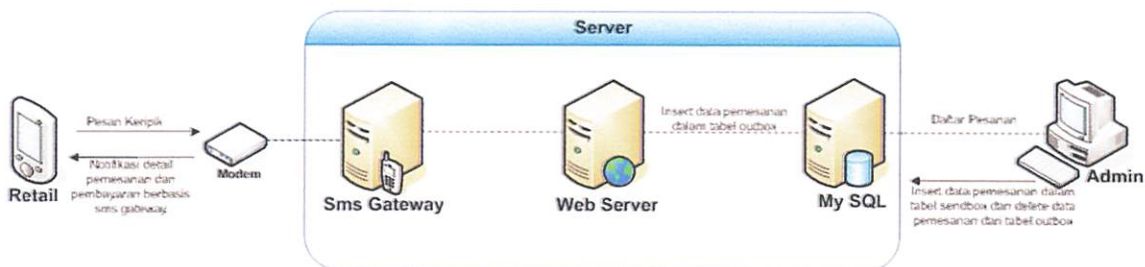
Pada bagian ini berisi tentang penjelasan mengenai perancangan sistem dan pembuatan aplikasi *Sales Force Automation* pemesanan keripik. Pada bagian ini memberikan beberapa gambaran secara umum tentang sistem yang akan dibuat. Tahapan perancangan merupakan langkah awal yang menentukan bagaimana membuat alur dan tampilan pada sistem yang akan dibuat. Langkah awal ini merupakan langkah yang sangat menentukan tingkat kesuksesan dalam pembuatan sistem.

Sistem yang akan dibuat adalah sistem pemesanan keripik, yang mana aplikasi ini nantinya terdiri dari 2 bagian yaitu aplikasi pada *mobile* android yang nantinya digunakan oleh *user / retail* untuk memesan keripik dan aplikasi server berbasis *web* yang akan digunakan oleh *admin*. Adapun gambaran mengenai aplikasi pemesanan keripik yang akan dibangun dapat dilihat pada gambar 3.3.



Gambar 3.3 *Arsitektur Sistem Pemesanan Produk*

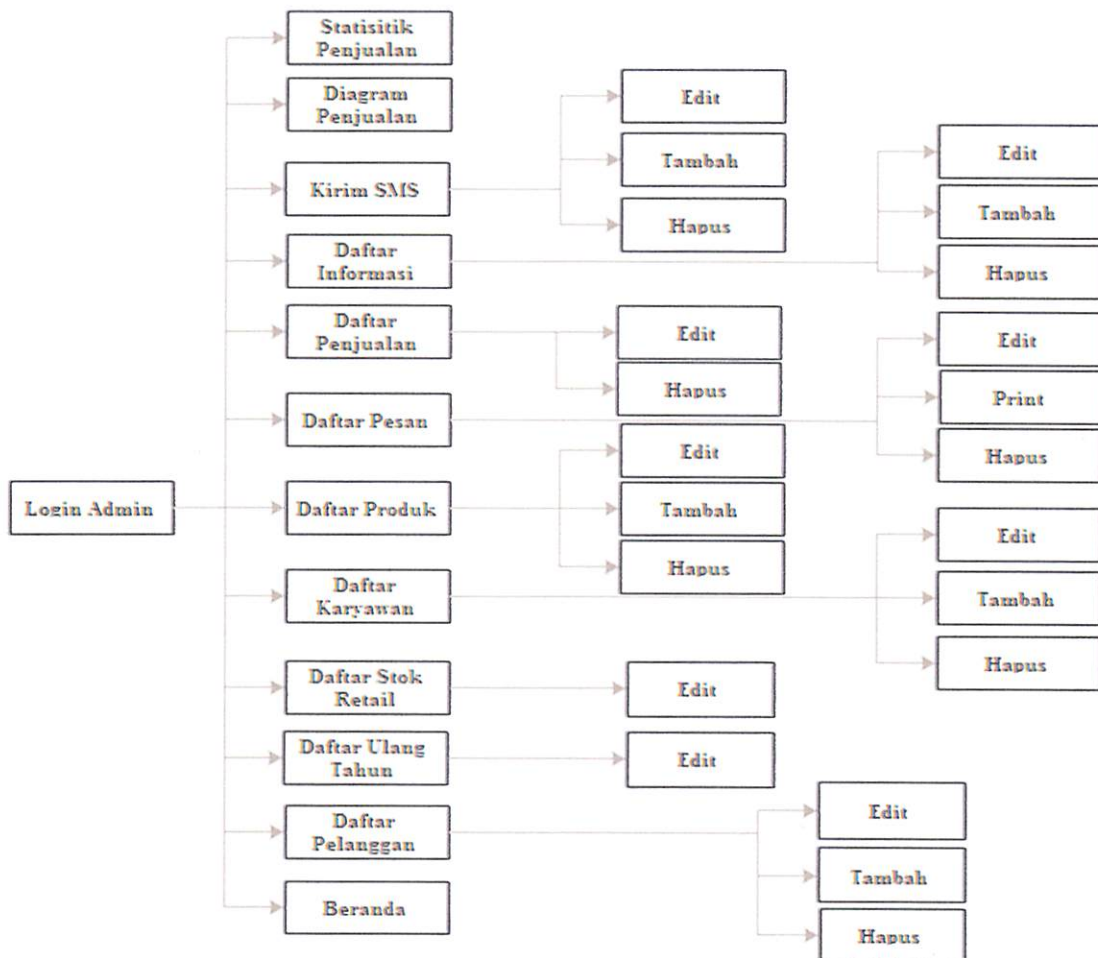
Pada gambar 3.3 diatas menjelaskan model sistem pemesanan produk, dimana *retail* melakukan pemesanan keripik, maka data pemesanan akan dikirim dan ditampilkan pada daftar pemesanan pada bagian *admin*, kemudian akan muncul detail pemesanan pada *retail*. Ketika *retail* melakukan konfirmasi pembayaran, maka *server* mengirim notifikasi pembayaran berbasis *sms gateway*. Adapun gambaran mengenai notifikasi pembayaran dapat dilihat pada gambar 3.4.



Gambar 3.4 *Arsitektur Sistem Notifikasi Pembayaran*

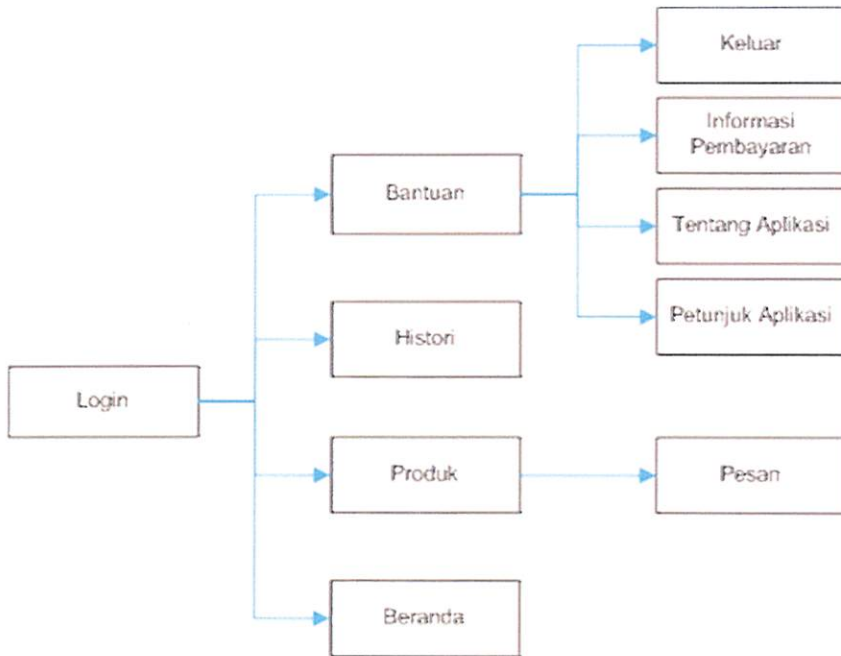
3.2.1 Rancangan Struktur Navigasi

Struktur navigasi merupakan struktur atau alur dari suatu program yang merupakan rancangan hubungan dari beberapa area yang berbeda yang dapat membantu dalam mengorganisasikan seluruh elemen dalam pembuatan program. Struktur navigasi merupakan hal yang dilakukan sebelum membuat suatu program. Pada gambar 3.5 dibawah dijelaskan bahwa pada halaman *admin* terdapat beberapa menu dengan beberapa aksi yang dapat dilakukan *admin* untuk melakukan pengolahan data yang tersedia.



Gambar 3.5 Struktur Navigasi Pada Aplikasi Halaman Admin

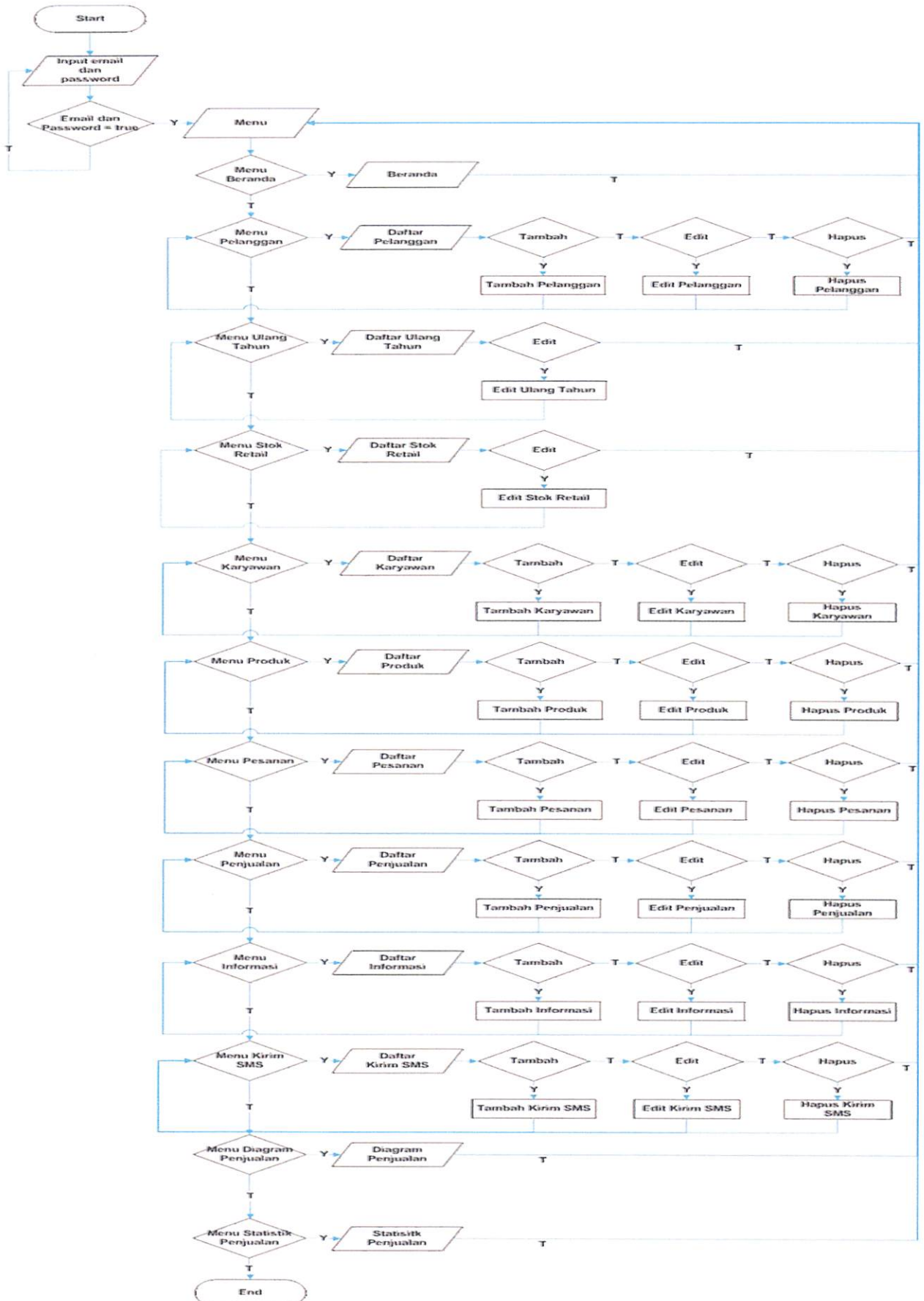
Pada gambar 3.6 dibawah dijelaskan bahwa pada aplikasi *mobile* terdapat beberapa menu untuk melakukan transaksi pemesanan keripik.



Gambar 3.6 Struktur Navigasi Pada Aplikasi Mobile

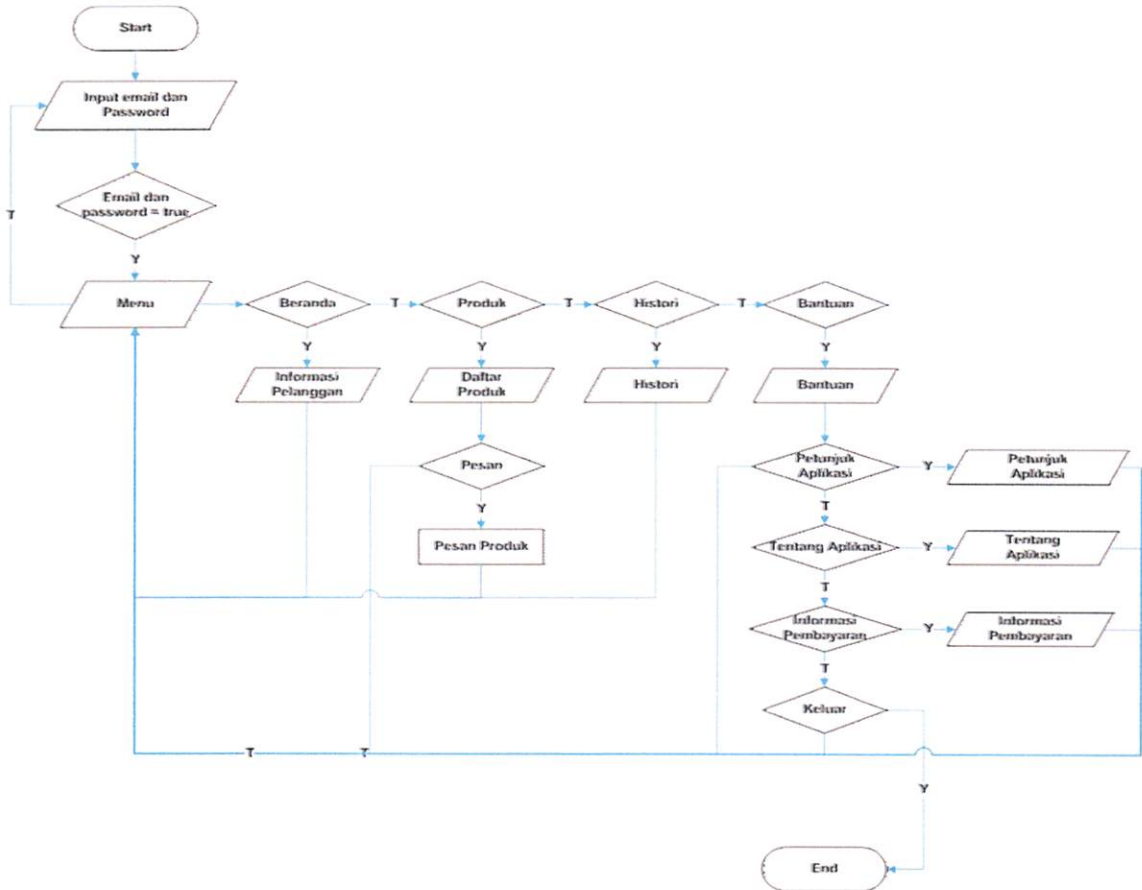
3.2.2 Rancangan Diagram Alir

Diagram alir (*Flowchart*) merupakan suatu gambaran secara grafik yang terdiri dari simbol-simbol dari algoritma dalam suatu program yang digunakan untuk menyatakan arah dari alur program. Pada gambar 3.7 dibawah menjelaskan tentang proses ketika menjalankan sistem pertama kali, dimana *administrator* harus menginputkan *email* dan *password*. Apabila *email* dan *password* bernilai *true*, maka *admin* dapat masuk pada halaman *admin* dan apabila bernilai *false*, maka akan dikembalikan pada *form login*. Pada halaman *admin* terdapat 12 menu pilihan, yaitu beranda, daftar pelanggan, daftar ulang tahun, daftar stok retail, daftar karyawan, daftar informasi, daftar produk, daftar pesanan, daftar penjualan, kirim sms, diagram penjualan dan statistik penjualan. *Administrator* dapat melakukan pengolahan data seperti menambah, mengedit dan menghapus data-data yang ada pada menu-menu tersebut.



Gambar 3.7 Flowchart Pada Aplikasi Server

Pada gambar 3.8 dibawah menjelaskan tentang proses pada aplikasi *mobile* ketika pelanggan ingin melakukan pemesanan keripik. Dalam aplikasi *mobile* terdapat 5 menu yaitu beranda, produk, histori, profil dan logout. Pada menu produk tersedia daftar produk dan tombol untuk melakukan pemesanan keripik. Daftar pemesanan akan ditampilkan pada menu histori.

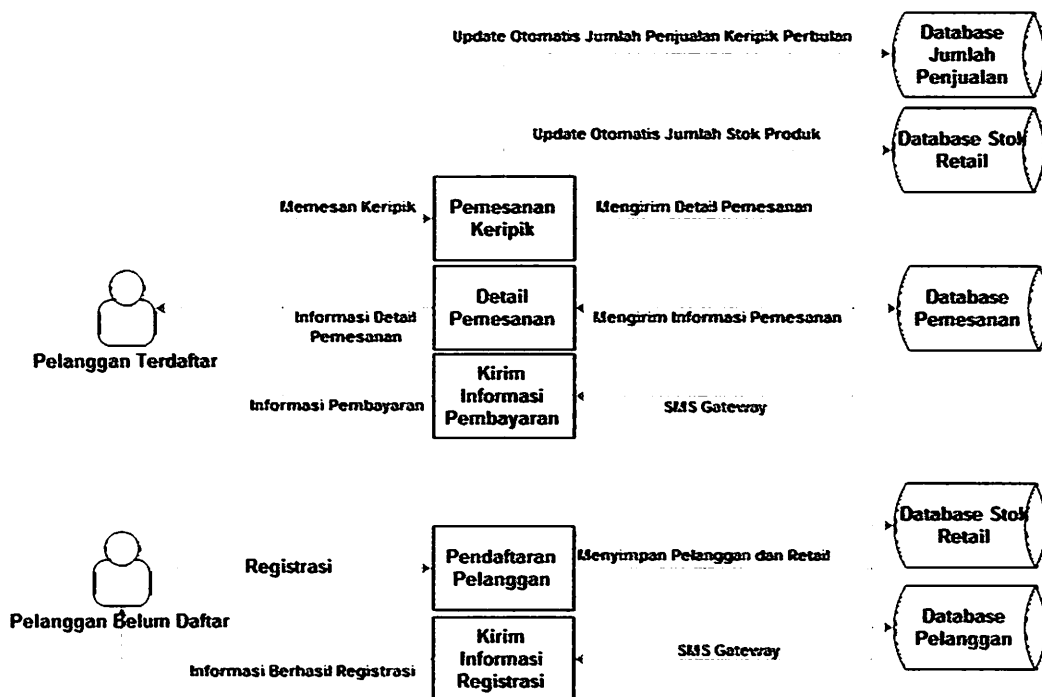


Gambar 3.8 Flowchart Pada Aplikasi Mobile

3.2.3 Rancangan Proses Pemesanan

Pada gambar 3.9 dibawah menjelaskan proses *Sales Force Automation* Pemesanan Keripik. Pelanggan yang belum terdaftar harus melakukan registrasi terlebih dahulu. Data-data registrasi pelanggan akan di simpan ke dalam *database* sebagai informasi data pelanggan dan data stok retail. Pelanggan yang sudah terdaftar akan menerima *sms* dari *server* berupa informasi berhasil melakukan registrasi, kemudian pelanggan dapat langsung melakukan *login* ke dalam aplikasi.

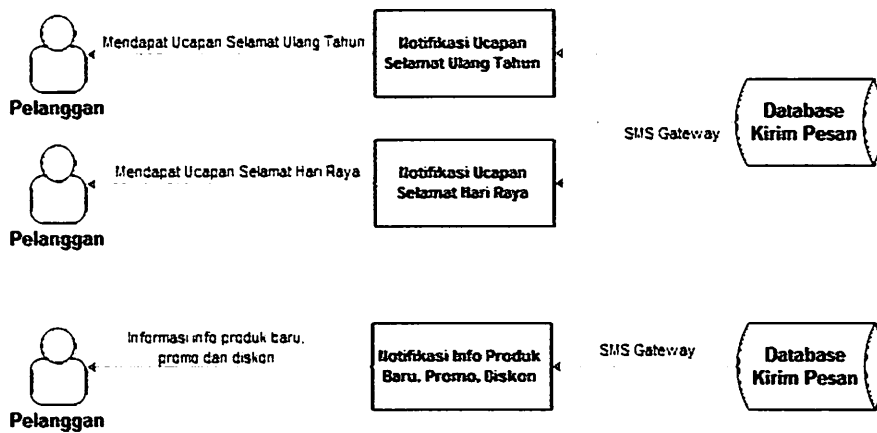
Data pemesanan keripik akan disimpan ke dalam *database* pada tabel pesanan. Pada saat selesai melakukan pemesanan, daftar pemesanan beserta jumlah bayar akan tampil pada aplikasi *mobile*. Pelanggan akan menerima pesan *sms* dari *server* berupa informasi pembayaran yang harus dilakukan. Stok retail secara otomatis juga akan berkurang sesuai pemesanan produk dan dalam jarak 1 hari, jumlah stok retail akan kembali pada stok semula. Berikut adalah proses *Sales Force Automation* Pemesanan Keripik yang dapat dilihat pada gambar 3.9.



Gambar 3.9 Proses *Sales Force Automation* Pemesanan Keripik

3.2.4 Rancangan Proses *Manual Notification*

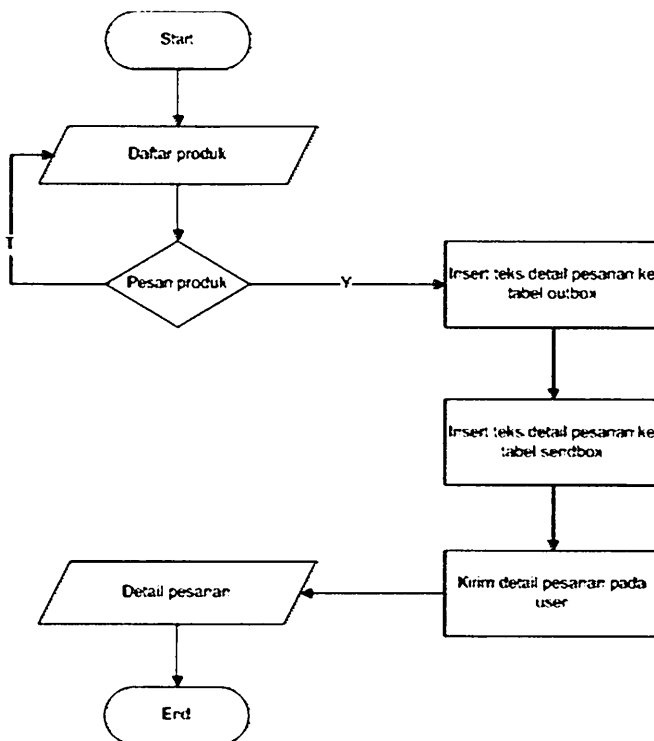
Pada gambar 3.10 menjelaskan proses pengiriman sms secara manual berbasis *SMS Gateway* kepada pelanggan mengenai info produk baru, promo bulanan, diskon, ucapan selamat ulang tahun dan ucapan selamat hari raya. Berikut adalah proses pengiriman notifikasi yang dapat dilihat pada gambar 3.10.



Gambar 3.10 Proses Manual Notification

3.2.5 Flowchart Auto Notification Pada Proses Pemesanan

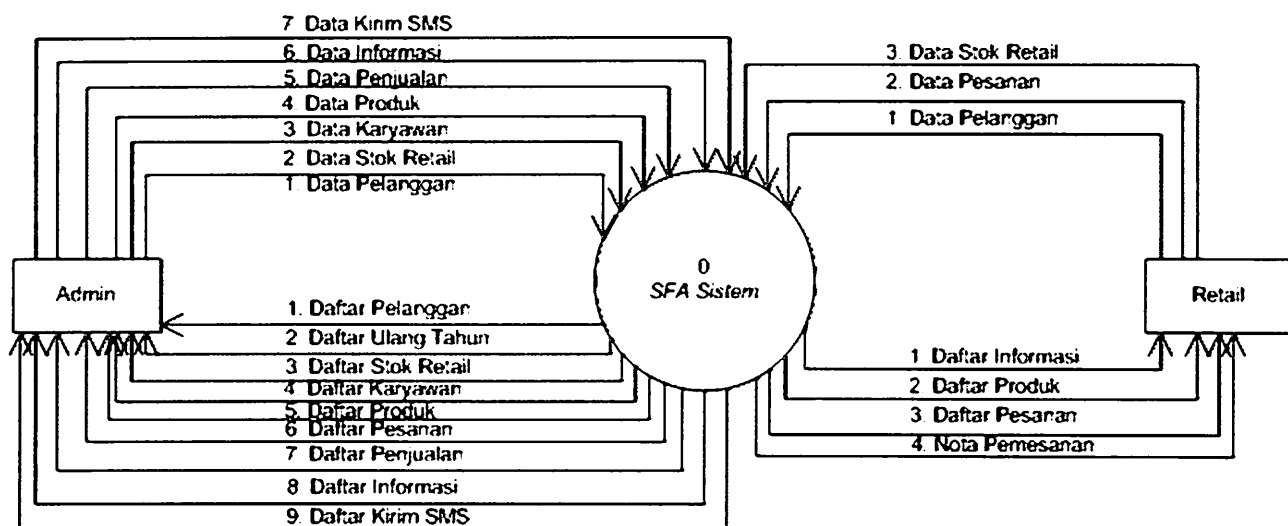
Pada gambar 3.11 menjelaskan proses *auto notification* berbasis *sms gateway* terhadap pelanggan ketika melakukan pemesanan keripik. Berikut adalah *flowchart auto notification* pada saat pelanggan melakukan pemesanan keripik yang dapat dilihat pada gambar 3.11.



Gambar 3.11 Flowchart Auto Notification Pada Proses Pemesanan

3.2.6 Rancangan Data Flow Diagram (DFD) Level 0

Data Flow Diagram level 0 adalah diagram pertama dari suatu Data Flow Diagram yang menggambarkan intensitas-intensitas yang berhubungan secara global. Berikut adalah rancangan data flow diagram yang dapat dilihat pada gambar 3.12.

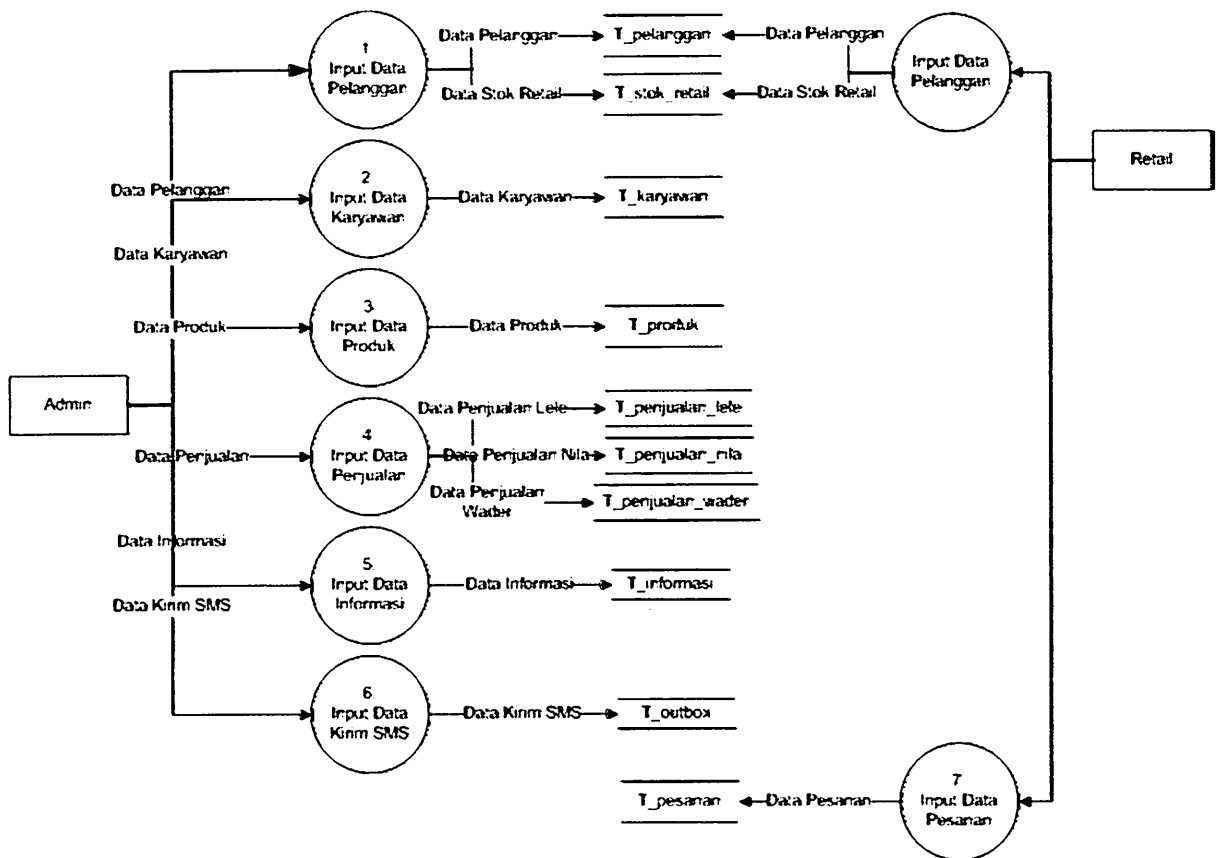


Gambar 3.12 Data Flow Diagram Level 0

Pada gambar 3.12 dijelaskan bahwa *admin* bertindak dalam menginputkan data pelanggan, data karyawan, data produk, data penjualan, data informasi dan data kirim sms. Sedangkan *Retail* bertindak dalam menginputkan data pelanggan, data pesanan dan data stok retail.

3.2.7 Rancangan Data Flow Diagram (DFD) Level 1

Tahap selanjutnya yaitu perancangan Data Flow Diagram Level 1 yang dapat dilihat pada gambar 3.13.

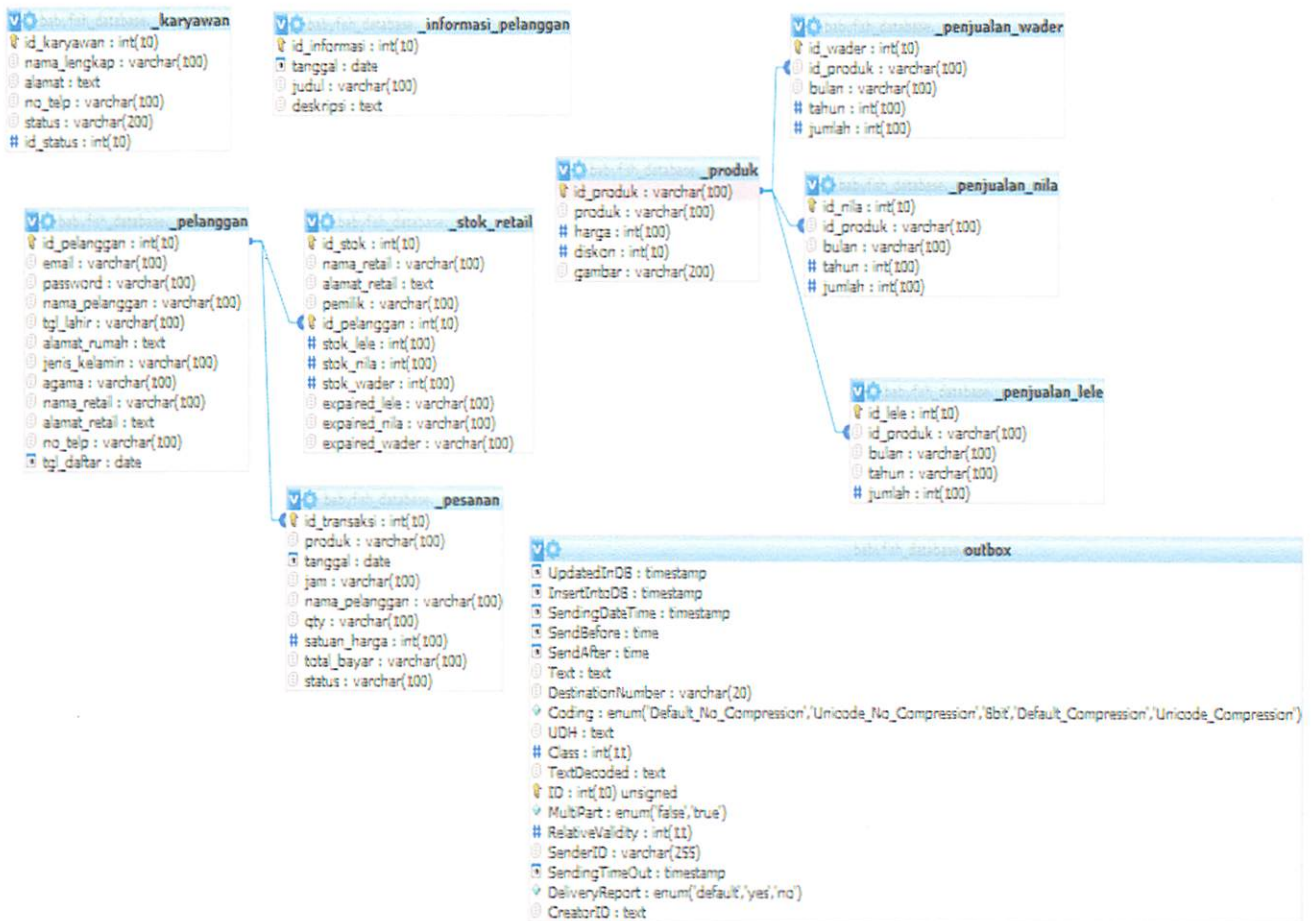


Gambar 3.13 Data Flow Diagram Level 1

Pada gambar 3.13 dijelaskan bahwa input data yang dilakukan *admin* dan *retail* akan disimpan pada tabel-tabel yang ada dalam *database*. Pada saat *admin* menginputkan data pelanggan, data akan disimpan pada tabel pelanggan dan tabel stok retail. Pada saat *admin* menginputkan data karyawan, data produk, data penjualan, data informasi dan data kirim sms, data akan disimpan pada tabel karyawan, tabel produk, tabel penjualan lele, tabel penjualan nila, tabel penjualan wader, tabel informasi dan tabel outbox. Pada saat *retail* menginputkan data pelanggan dan data pesanan, data akan disimpan pada tabel pelanggan, tabel stok retail dan tabel pesanan.

3.2.8 Entyty Relation Diagram (ERD)

Entyty Relation Diagram merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. Berikut adalah model struktur data dan hubungan antara data yang dapat dilihat pda gambar 3.14.



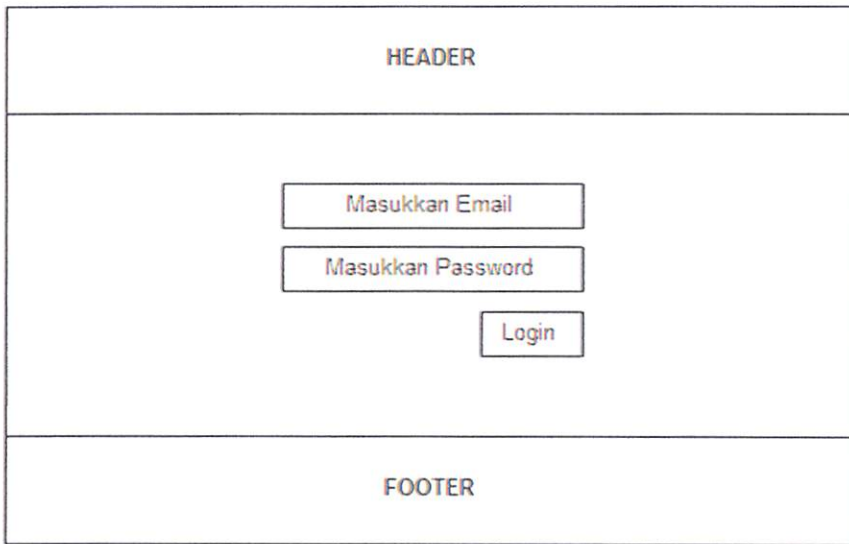
Gambar 3.14 Entity Relation Diagram

3.3 Desain Interface Sistem

Pada tahap selanjutnya adalah membuat rancangan desain *interface* sistem dalam bentuk prototipe.

3.3.1 Halaman Login Admin

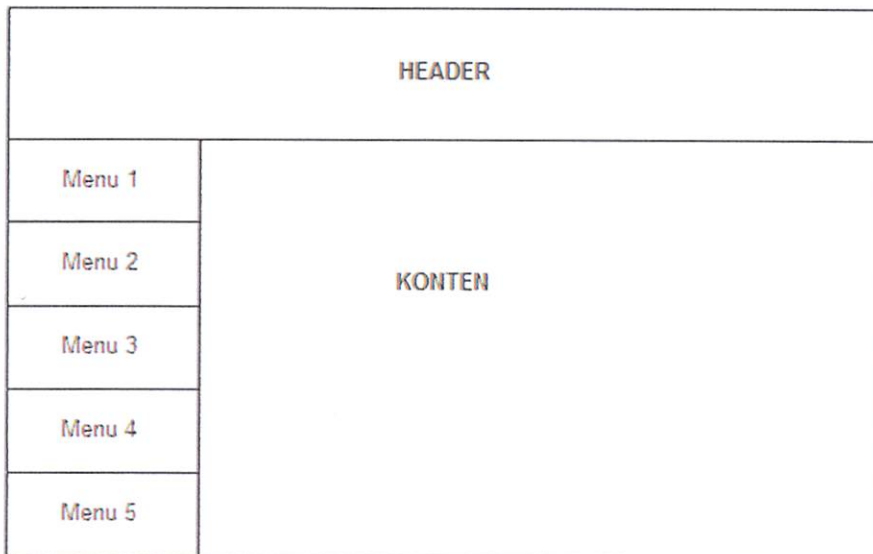
Halaman login admin merupakan halaman pada bagian *web* yang digunakan oleh *admin* untuk masuk ke halaman *admin*. Pada *login admin* terdapat dua *form* yang harus diisi untuk dapat masuk ke halaman *admin* yaitu *email* dan *password*. Berikut adalah prototipe untuk desain halaman *login admin* yang dapat dilihat pada gambar 3.15.



Gambar 3.15 *Desain Prototipe Halaman Login Admin*

3.3.2 Halaman Menu *Administrator*

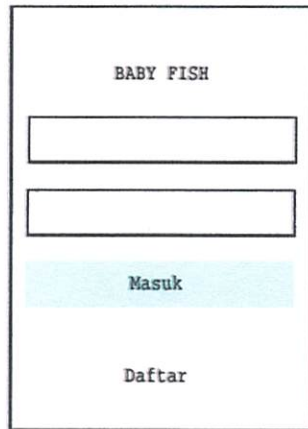
Halaman menu *administrator* merupakan halaman khusus untuk *administrator* dalam melakukan manajemen data. Pada halaman ini berisi menu–menu yang akan memudahkan *admin* untuk mengolah seluruh data yang ada, seperti menambah, merubah, dan menghapus data yang ada. Berikut adalah prototipe untuk desain halaman *admin* yang dapat dilihat pada gambar 3.16.



Gambar 3.16 *Desain Prototipe Halaman Menu Administrator*

3.3.3 Form Login Android Mobile

Form login *android mobile* merupakan form yang digunakan user atau pelanggan untuk masuk ke halaman menu utama pada aplikasi pesanan keripik. Terdapat dua *form* yang harus diisi yaitu *email* dan *password*. Berikut adalah prototipe untuk desain *form login* pada aplikasi *mobile* yang dapat dilihat pada gambar 3.17.

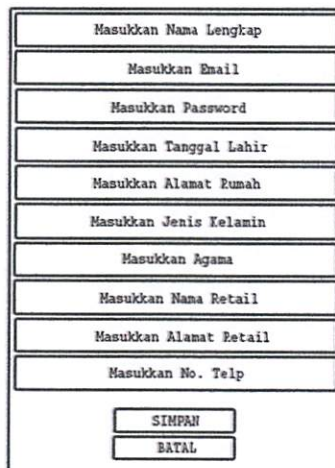


The image shows a mobile login form prototype. At the top, it says "BABY FISH". Below that are two empty rectangular input fields. Under the second field is a light blue button labeled "Masuk". At the bottom of the form is a text link labeled "Daftar".

Gambar 3.17 Desain Prototipe Form Login Android Mobile

3.3.4 Form Registrasi Android Mobile

Untuk pelanggan yang belum memiliki akun, dapat melakukan pendaftaran terlebih dahulu dengan memilih *button* daftar yang terletak di bawah *button login*. Terdapat beberapa *form* yang harus diisi pada *form* registrasi untuk melengkapi beberapa data diri pelanggan. Berikut adalah prototipe untuk desain *form* registrasi yang dapat dilihat pada gambar 3.18.



The image shows a mobile registration form prototype. It consists of a vertical stack of eleven input fields, each with a label: "Masukkan Nama Lengkap", "Masukkan Email", "Masukkan Password", "Masukkan Tanggal Lahir", "Masukkan Alamat Rumah", "Masukkan Jenis Kelamin", "Masukkan Agama", "Masukkan Nama Retail", "Masukkan Alamat Retail", and "Masukkan No. Telp". At the bottom of the form are two buttons: "SIMPAN" and "BATAL".

Gambar 3.18 Desain Prototipe Form Registrasi

3.3.5 Tampilan Menu Beranda

Berikut adalah prototipe untuk desain halaman menu beranda yang dapat dilihat pada gambar 3.19.

Beranda Produk Histori Bantuan
Petunjuk Aplikasi
Tentang Aplikasi
Informasi Pembayaran

Gambar 3.19 *Desain Prototipe Android Mobile Menu Beranda*

Pada menu beranda berisi berupa informasi – informasi berupa promosi, pengumuman penting dan kumpulan berita yang berkaitan dengan usaha bisnis *Baby Fish*.

3.3.6 Tampilan Menu Produk

Berikut adalah prototipe untuk desain halaman menu produk yang dapat dilihat pada gambar 3.20.

Beranda Produk Histori Bantuan	Form Pesan
Nama Produk Harga	Nama Produk
Nama Produk Harga	Quantity
Nama Produk Harga	Pesan

Gambar 3.20 *Desain Prototipe Android Mobile Menu Produk*

3.3.7 Tampilan Menu Histori

Berikut adalah prototipe untuk desain halaman menu beranda yang dapat dilihat pada gambar 3.21.

Beranda Produk Histori Bantuan	
Histori 1 Tanggal Id Transaksi	Status
Histori 2 Tanggal Id Transaksi	Status
Histori 3 Tanggal Id Transaksi	Status

Gambar 3.21 *Desain Prototipe Android Mobile Menu History*

Pada menu history berisi tentang daftar pemesanan keripik yang pernah dilakukan, seperti tanggal pemesanan, informasi pesanan, id transaksi dan status pemesanan.

3.3.8 Tampilan Menu Bantuan

Berikut adalah prototipe untuk desain halaman menu bantuan yang dapat dilihat pada gambar 3.22.

Beranda Produk Histori Bantuan
Petunjuk Aplikasi
Tentang Aplikasi
Informasi Pembayaran
Keluar

Gambar 3.22 *Desain Prototipe Android Mobile Menu Bantuan*

Pada menu bantuan berisi tentang petunjuk aplikasi, tentang aplikasi, informasi pembayaran dan menu keluar dari aplikasi.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Sistem

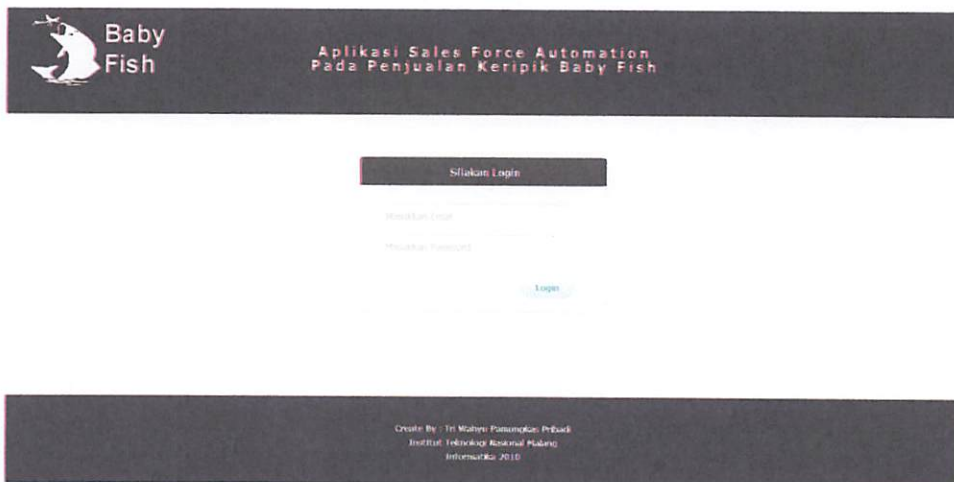
Tahap implementasi sistem merupakan tahapan yang dilakukan dengan menerapkan hasil desain yang telah dibuat ke dalam bahasa pemrograman berbasis *web* dan *android*, yaitu PHP dan Java sehingga prosedur–prosedur yang telah dibuat dapat menghasilkan suatu keluaran seperti yang diharapkan.

4.1.1 Bagian Administrasi

Halaman administrasi merupakan tampilan yang digunakan untuk seorang administrator. Pada halaman ini berisi daftar menu–menu yang digunakan untuk mengolah data yang ada, seperti halnya dengan menambah, merubah dan menghapus data yang ada.

4.1.1.1 Tampilan Halaman *Login*

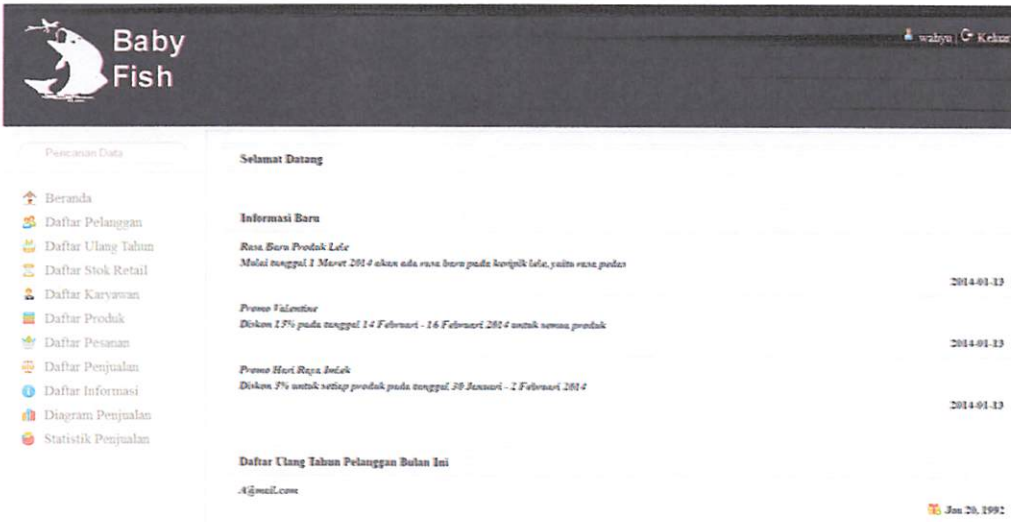
Halaman *login* merupakan halaman yang pertama kali muncul pada saat aplikasi *web* dijalankan, dimana pada halaman ini terdapat *form* yang harus diisi dengan *email* dan *password administrator* agar dapat masuk ke dalam halaman administrasi. Jika *email* dan *password* benar maka akan menuju halaman administrasi. Berikut adalah gambar desain halaman *login* yang dapat dilihat pada gambar 4.1.



Gambar 4.1 Tampilan Halaman *Login*

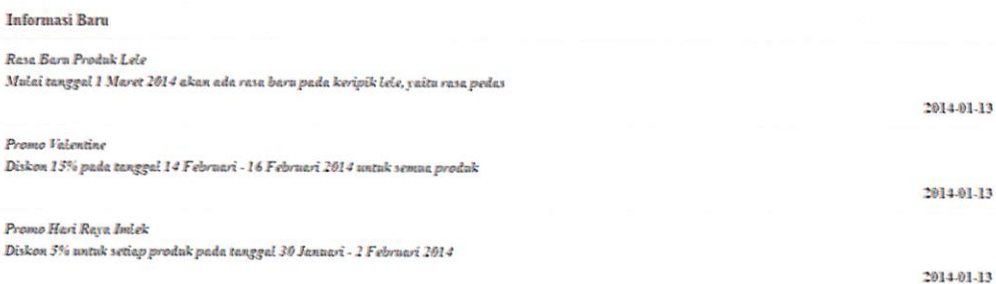
4.1.1.2 Tampilan Halaman Beranda

Halaman beranda merupakan halaman pertama yang ditampilkan ketika pertama kali masuk ke halaman administrasi. Berikut adalah tampilan halaman beranda pada halaman administrasi yang dapat dilihat pada gambar 4.2.



Gambar 4.2 Tampilan Halaman Beranda

Dalam halaman beranda terdapat beberapa informasi dari beberapa menu dari halaman administrasi yang ditampilkan sebagai informasi *up-to-date*, seperti informasi yang ditampilkan pada pelanggan, informasi daftar ulang tahun pelanggan bulan ini, informasi penjualan produk tahun ini beserta grafiknya. Berikut adalah tampilan informasi pelanggan yang dapat dilihat pada gambar 4.3.



Gambar 4.3 Tampilan Informasi Baru Pada Halaman Beranda

Selanjutnya adalah tampilan informasi daftar ulang tahun pada bulan ini dapat dilihat pada gambar 4.4.

Daftar Ulang Tahun Pelanggan Bulan Ini	
Tri Wahyu Pamungkas Prabedi	Jan 15, 1992
Andri Retno Hasono	Jan 2, 1991
Lutfi Akid Priyono	Jan 21, 1992

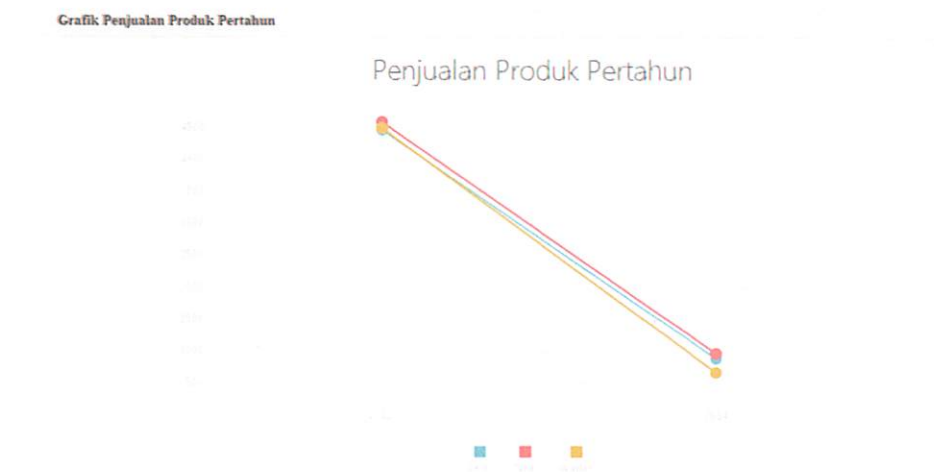
Gambar 4.4 Tampilan Informasi Daftar Ulang Tahun Bulan Ini Pada Halaman Beranda

Selanjutnya adalah tampilan informasi jumlah penjualan produk pada tahun ini yang dapat dilihat pada gambar 4.5.

Jumlah Penjualan Tahun Ini	
Penjualan Keripik Lete : 835	
Penjualan Keripik Nila : 969	
Penjualan Keripik Wader : 665	

Gambar 4.5 Tampilan Informasi Jumlah Penjualan Produk Tahun Ini Pada Halaman Beranda

Selanjutnya adalah tampilan grafik jumlah penjualan produk pertahun yang dapat dilihat pada gambar 4.6.



Gambar 4.6 Tampilan Grafik Jumlah Penjualan Produk Pertahun Pada Halaman Beranda

4.1.1.3 Halaman Daftar Pelanggan

Halaman daftar pelanggan merupakan halaman yang menampilkan data-data pelanggan yang sudah terdaftar. Pada menu ini *admin* juga dapat menambah data pelanggan baru, merubah dan menghapus data-data pelanggan yang telah terdaftar. Berikut adalah tampilan daftar pelanggan yang dapat dilihat pada gambar 4.7.

No	Nama Pelanggan	Email	Alamat Rumah	Jenis Kelamin	Agama	No. Telp	Terdaftar	Aksi
1	Indah Widha Christyana Widan	ttoto12@gmail.com	Jl. Selin Mandaha No 90 RT 02W 02 01 Sumber Pucung 3.Calang	Perempuan	Kristen	03140907987	2014-01-01	✎ ✕
2	M. Teufique Fawid	mtda_gombang@yahoo.co.id	Jl. Slamet Riyadi 012 RT 02W 02 01 Gombang Kec. Lerok Sambung Kec. Lerok Wam. 3.Calang	Laki - Laki	Islam	081541789913	2014-01-13	✎ ✕
3	Luhul Akhadin Puyena	lhu0123@gmail.com	Jl. Kesubrang No 58 RT 02W 04-09 Kawar Kec. Dano 3.Calang	Laki - Laki	Islam	081789017967	2014-01-01	✎ ✕
4	Andri Rahen Wawono	andri15@gmail.com	Jl. Bantulan I No 12 RT 02W 14 01 Kec. Dano 3.Calang	Laki - Laki	Islam	08178908874	2014-01-01	✎ ✕
5	Tu Wahyu Panglipis Petrah	tuwx735@gmail.com	Jl. Raya Pesisiran 0118 RT 02W 03 01 Kec. Kesunggan 3.Calang	Laki - Laki	Islam	08171478739	2014-01-01	✎ ✕

Gambar 4.7 Tampilan Halaman Daftar Pelanggan

4.1.1.4 Halaman Tambah Data Pelanggan

Halaman ini merupakan halaman yang digunakan *admin* untuk menambah daftar pelanggan baru. Berikut adalah tampilan halaman tambah pelanggan yang dapat dilihat pada gambar 4.8.

Input Pelanggan Baru

Nama Pelanggan Masukkan Nama Pelanggan

Email Masukkan Alamat Email

Password Masukkan Password

Tgl Lahir Masukkan Tgl Lahir

Jenis Kelamin Laki-Laki Perempuan

Agama -Agama-

Alamat Rumah

Nama Retail Masukkan Nama Retail

Alamat Retail

No Telp Masukkan Nomor Telepon

Gambar 4.8 Tampilan Halaman Tambah Pelanggan

4.1.1.5 Halaman Edit Data Pelanggan

Halaman ini merupakan halaman yang digunakan *admin* untuk merubah data pelanggan yang ada. Berikut adalah tampilan halaman edit pelanggan yang dapat dilihat pada gambar 4.9.

Gambar 4.9 Tampilan Halaman Edit Pelanggan

4.1.1.6 Halaman Daftar Ulang Tahun

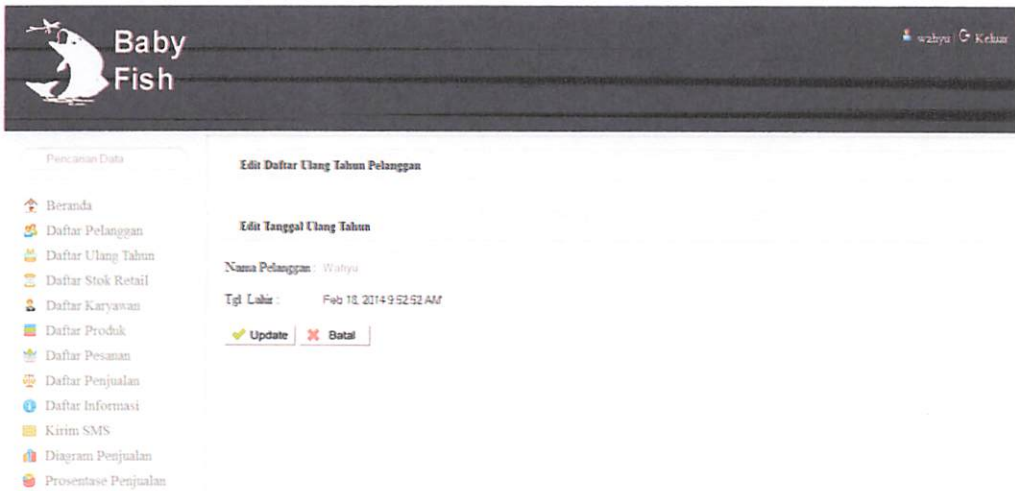
Halaman daftar ulang tahun merupakan halaman yang menampilkan tanggal lahir pelanggan yang akan digunakan sistem untuk memberi ucapan selamat ketika pelanggan ulang tahun. Berikut adalah tampilan halaman daftar ulang tahun pelanggan yang dapat dilihat pada gambar 4.10

No	Nama Pelanggan	Tgl. Lahir	Aksi
1	Indah Widhi Chrysyan Wulan	Dec 19, 1985	
2	M. Taufiqur Razihi	Mar 9, 1991	
3	Luhaf Aladin Poyono	Jan 21, 1992	
4	Audin Zetson Wawana	Jan 2, 1991	
5	Ta Wabyu Pamungkas Pebuali	Jan 11, 1992	

Gambar 4.10 Tampilan Halaman Daftar Ulang Tahun

4.1.1.7 Halaman Edit Ulang Tahun

Halaman edit ulang tahun merupakan halaman yang digunakan *admin* untuk merubah data tanggal lahir pelanggan. Berikut adalah tampilan halaman edit ulang tahun yang dapat dilihat pada gambar 4.11.



Gambar 4.11 Tampilan Halaman Edit Ulang Tahun

4.1.1.8 Halaman Daftar Stok Retail

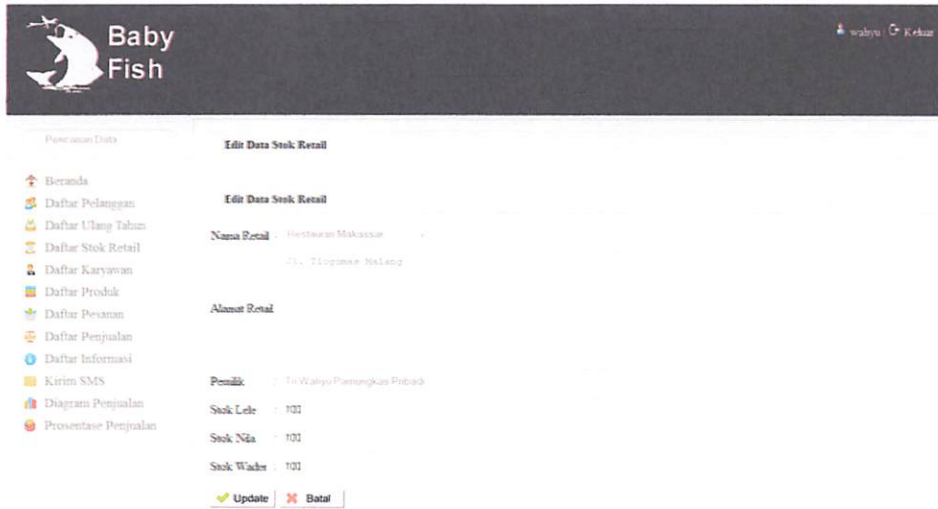
Halaman daftar stok retail merupakan halaman yang menampilkan stok produk untuk retail yang sudah terdaftar. Pada menu ini *admin* dapat menambah, merubah dan menghapus stok-stok produk yang telah ditentukan. Berikut adalah tampilan daftar stok retail yang dapat dilihat pada gambar 4.12.



Gambar 4.12 Tampilan Halaman Daftar Stok Retail

4.1.1.9 Halaman Edit Stok Retail

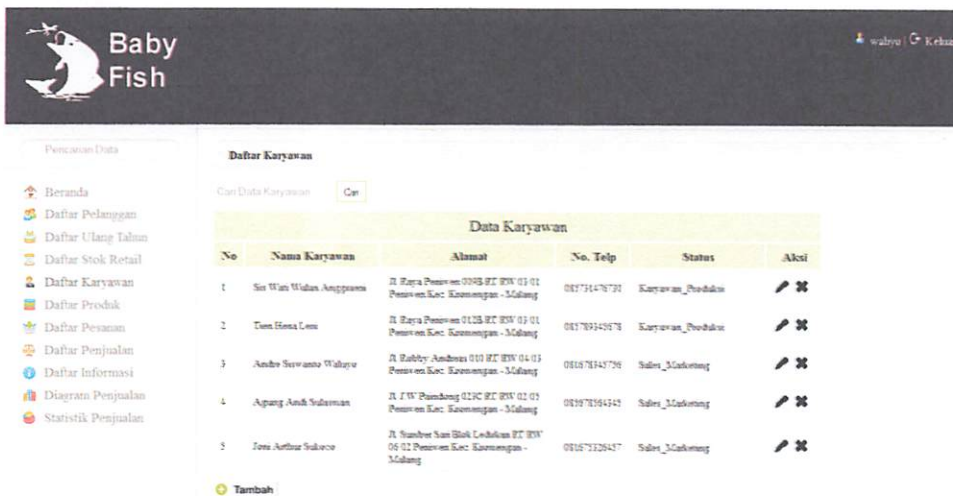
Halaman ini merupakan halaman yang digunakan *admin* untuk merubah jumlah stok retail pelanggan yang disediakan. Berikut adalah tampilan halaman edit stok retail yang dapat dilihat pada gambar 4.13.



Gambar 4.13 Tampilan Halaman Edit Stok Retail

4.1.1.10 Halaman Daftar Karyawan

Halaman daftar karyawan merupakan halaman yang menampilkan semua data-data karyawan yang ada. Pada menu ini *admin* dapat menambah, merubah dan menghapus data karyawan yang ada. Berikut adalah tampilan daftar karyawan yang dapat dilihat pada gambar 4.14.



Gambar 4.14 Tampilan Halaman Daftar Karyawan

4.1.1.11 Halaman Tambah Data Karyawan

Halaman ini merupakan halaman yang digunakan *admin* untuk menambah daftar karyawan baru. Berikut adalah tampilan halaman tambah karyawan yang dapat dilihat pada gambar 4.15.

The screenshot shows the 'Tambah Karyawan' page. On the left is a sidebar with a 'Pencarian Data' search bar and a list of menu items: Beranda, Daftar Pelanggan, Daftar Ulang Tahun, Daftar Stok Retail, Daftar Karyawan, Daftar Produk, Daftar Pesanan, Daftar Penjualan, Daftar Informasi, Diagram Penjualan, and Statistik Penjualan. The main content area is titled 'Tambah Karyawan' and contains the following form fields:

- Input Karyawan Baru**
- Nama Pelanggan:** Masukkan Nama Karyawan
- Alamat Karyawan:** Masukkan Alamat Lengkap
- No. Telp:** Masukkan Nomor Telepon
- Status:** -Status Karyawan-

At the bottom of the form are two buttons: a green 'Simpan' button and a red 'Batal' button.

Gambar 4.15 Tampilan Halaman Tambah Karyawan

4.1.1.12 Halaman Edit Data Karyawan

Halaman ini merupakan halaman yang digunakan *admin* untuk merubah data karyawan yang ada. Berikut adalah tampilan halaman edit data karyawan yang dapat dilihat pada gambar 4.16.

The screenshot shows the 'Edit Data Karyawan' page. On the left is a sidebar with a 'Pencarian Data' search bar and a list of menu items: Beranda, Daftar Pelanggan, Daftar Ulang Tahun, Daftar Stok Retail, Daftar Karyawan, Daftar Produk, Daftar Pesanan, Daftar Penjualan, Daftar Informasi, Diagram Penjualan, and Statistik Penjualan. The main content area is titled 'Edit Data Karyawan' and contains the following form fields:

- Edit Data Karyawan**
- Nama Karyawan:** Andre Sismanti Waluyo
- Alamat Karyawan:** Jl. Robby Andreak 010 RT/ RW 04/03 Pentawen Eco. Kruangan - Malang
- No. Telp:** 081678145796
- Status:** Sales_Marketing

At the bottom of the form are two buttons: a green 'Update' button and a red 'Batal' button.

Gambar 4.16 Tampilan Halaman Edit Karyawan

... ..

... ..

... ..



... ..

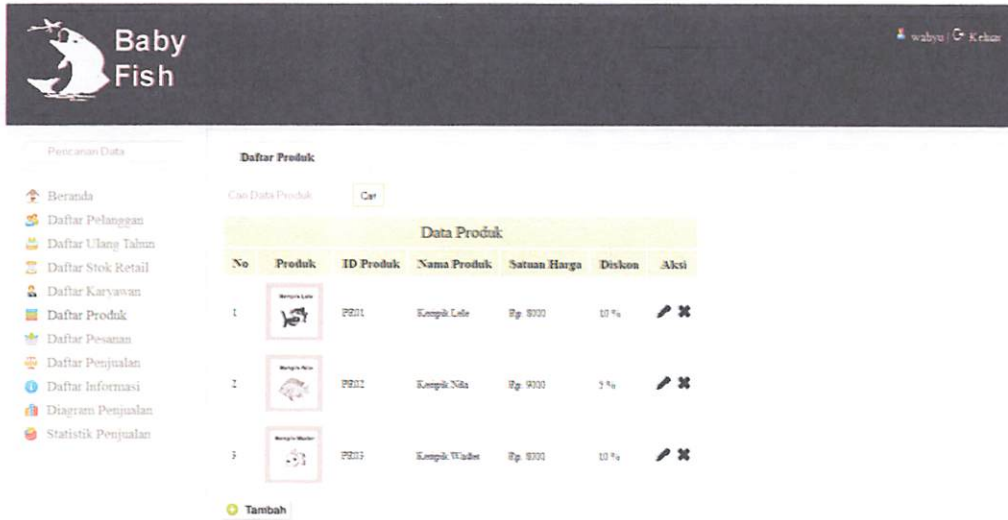
... ..

... ..



4.1.1.13 Halaman Daftar Produk

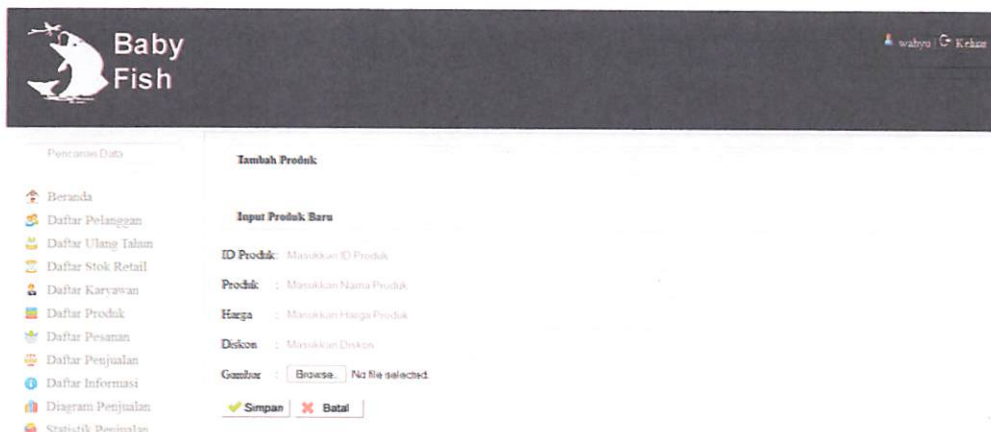
Halaman daftar produk merupakan halaman yang menampilkan daftar produk yang ada. Pada menu ini *admin* dapat menambah, merubah dan menghapus data produk yang ada. Berikut adalah tampilan daftar produk yang dapat dilihat pada gambar 4.17.



Gambar 4.17 Tampilan Halaman Daftar Produk

4.1.1.14 Halaman Tambah Produk

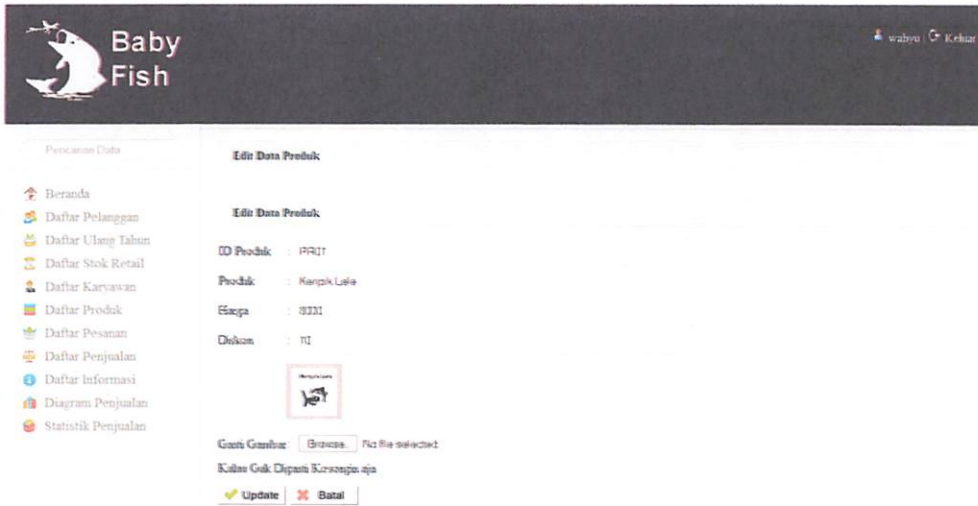
Halaman ini merupakan halaman yang digunakan *admin* untuk menambah daftar produk baru. Berikut adalah tampilan halaman tambah produk yang dapat dilihat pada gambar 4.18.



Gambar 4.18 Tampilan Halaman Tambah Produk

4.1.1.15 Halaman Edit Produk

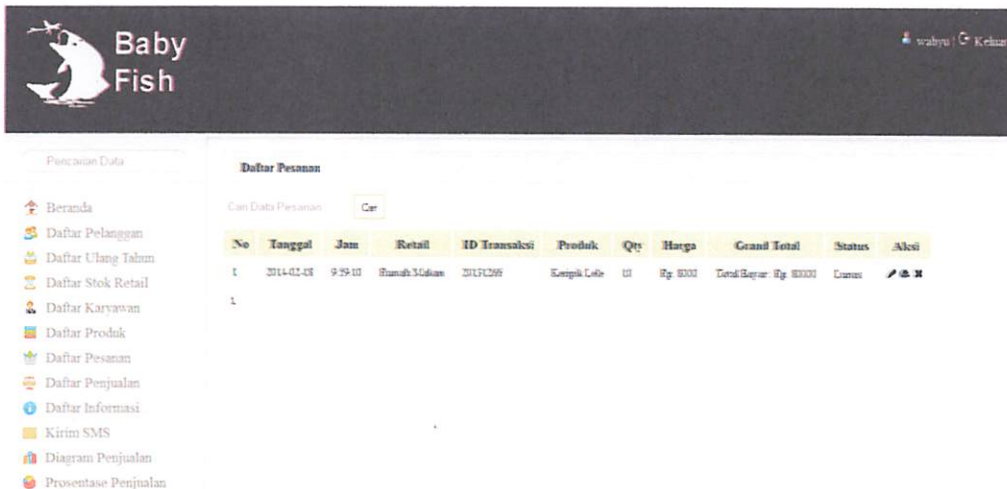
Halaman ini merupakan halaman yang digunakan *admin* untuk merubah data produk yang ada. Berikut adalah tampilan halaman edit produk yang dapat dilihat pada gambar 4.19.



Gambar 4.19 Tampilan Halaman Edit Produk

4.1.1.16 Halaman Daftar Pesan

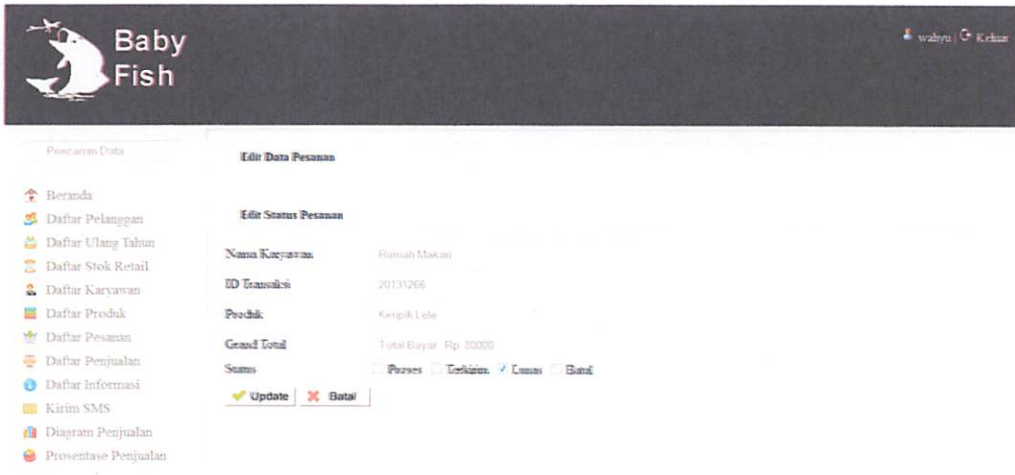
Halaman daftar pesan merupakan halaman yang menampilkan daftar pesanan produk yang ada. Pada halaman ini *admin* hanya dapat merubah status pesanan dan menghapus daftar pesanan. Berikut adalah tampilan daftar pesan yang dapat dilihat pada gambar 4.20.



Gambar 4.20 Tampilan Halaman Daftar Pesanan

4.1.1.17 Halaman Edit Status Pesan

Halaman ini merupakan halaman yang digunakan *admin* untuk merubah status pesanan. Berikut adalah tampilan halaman edit status pesan yang dapat dilihat pada gambar 4.21.



Gambar 4.21 Tampilan Edit Status Pesan

4.1.1.18 Halaman Daftar Penjualan

Halaman daftar penjualan merupakan halaman yang menampilkan jumlah penjualan produk perbulan. Berikut adalah tampilan daftar penjualan yang dapat dilihat pada gambar 4.22.

The screenshot shows the 'Daftar Penjualan' interface. On the left is a sidebar with navigation items like 'Beranda', 'Daftar Pelanggan', 'Daftar Ulang Tahun', 'Daftar Stok Retail', 'Daftar Karyawan', 'Daftar Produk', 'Daftar Pesanan', 'Daftar Penjualan', 'Daftar Informasi', 'Diagram Penjualan', and 'Statistik Penjualan'. The main content area is titled 'Daftar Penjualan' and displays a table with sales data for three products: Keripik Lele, Keripik Nila, and Keripik Wader.

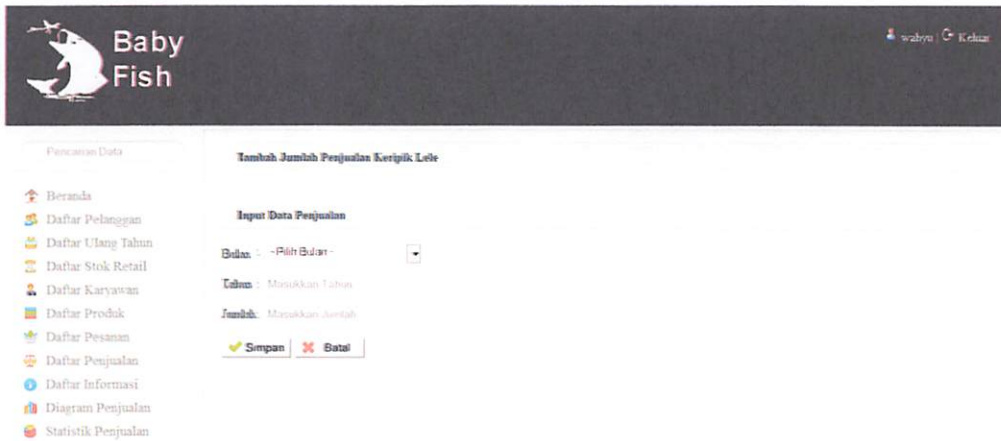
Keripik Lele					Keripik Nila					Keripik Wader				
No	Bulan	Tahun	Jumlah	Aksi	No	Bulan	Tahun	Jumlah	Aksi	No	Bulan	Tahun	Jumlah	Aksi
1	Februari	2014	479		1	Februari	2014	479		1	Februari	2014	520	
2	Januari	2014	490		2	Januari	2014	480		2	Januari	2014	443	
3	Desember	2013	372		3	Desember	2013	372		3	Desember	2013	320	
4	November	2013	373		4	November	2013	320		4	November	2013	380	
5	Oktober	2013	423		5	Oktober	2013	423		5	Oktober	2013	340	
6	September	2013	490		6	September	2013	320		6	September	2013	380	
7	Agustus	2013	420		7	Agustus	2013	423		7	Agustus	2013	350	
8	Juli	2013	443		8	Juli	2013	423		8	Juli	2013	320	
9	Juni	2013	340		9	Juni	2013	320		9	Juni	2013	443	
10	Mei	2013	343		10	Mei	2013	423		10	Mei	2013	380	

At the bottom of each table section, there are pagination controls: '1 2', 'Tambah', and 'Tampilkan'.

Gambar 4.22 Tampilan Daftar Penjualan

4.1.1.19 Halaman Tambah Data Penjualan

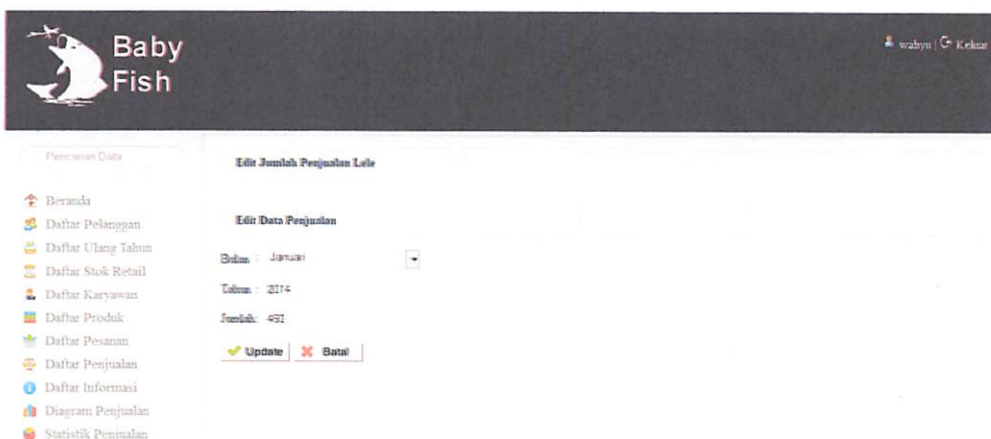
Halaman ini merupakan halaman yang digunakan *admin* untuk menambah data penjualan perbulan. Berikut adalah tampilan halaman tambah data penjualan yang dapat dilihat pada gambar 4.23.



Gambar 4.23 Tampilan Tambah Data Penjualan

4.1.1.20 Halaman Edit Data Penjualan

Halaman ini merupakan halaman yang digunakan *admin* untuk merubah data penjualan yang ada. Berikut adalah tampilan halaman edit data penjualan yang dapat dilihat pada gambar 4.24.



Gambar 4.24 Tampilan Edit Data Penjualan

4.1.1.21 Halaman Daftar Informasi

Halaman daftar informasi merupakan halaman yang menampilkan daftar informasi pelanggan. Berikut adalah tampilan daftar informasi yang dapat dilihat pada gambar 4.25.

No	Tanggal Posting	Judul	Deskripsi	Aksi
1	2014-01-13	Rasa Baru Produk Lela	Mula tanggal 13Januari 2014 akan ada rasa baru produk lelak lela, yaitu rasa pedas	
2	2014-01-13	Promo Valentine	Diskon 10% pada tanggal 14 Februari - 15 Februari 2014 untuk semua produk	
3	2014-01-13	Promo Hari Raya Idul Fitri	Diskon 5% untuk setiap produk pada tanggal 21 Januari - 2 Februari 2014	
4	2014-01-13	Promo Bulan Januari	Diskon 10% untuk setiap produk mulai tanggal 13 Januari - 31 Januari 2014	
7	2013-02-20	Produk Baru	Ada 500 akan melancarkan produk baru yaitu lelak lela dengan 2 rasa, yaitu rasa pedas dan rasa manis tanggal 21 Januari 2014	
6	2013-02-20	Launching 2 Rasa Lemak Lela	Akan diluncurkan rasa pedas dan rasa manis produk lelak lela tanggal 21 Januari 2014	

Gambar 4.25 Tampilan Daftar Informasi

4.1.1.22 Halaman Tambah Informasi

Halaman ini merupakan halaman yang digunakan *admin* untuk menambah daftar informasi pelanggan. Berikut adalah tampilan halaman tambah informasi yang dapat dilihat pada gambar 4.26.

Gambar 4.26 Tampilan Tambah Informasi

4.1.1.23 Halaman Edit Informasi

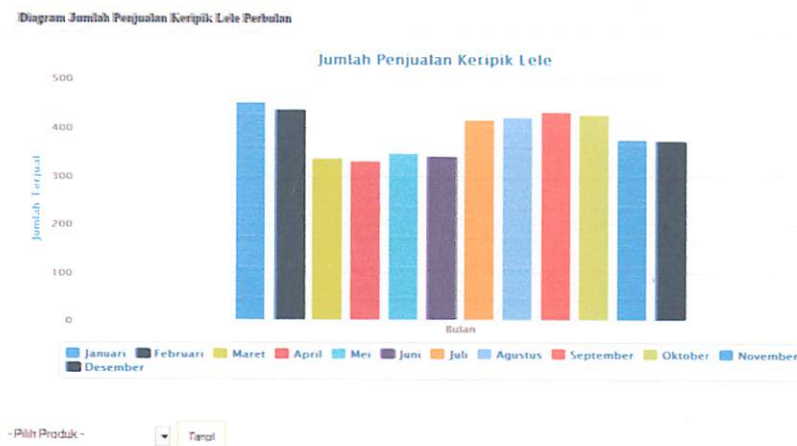
Halaman ini merupakan halaman yang digunakan *admin* untuk merubah daftar informasi pelanggan. Berikut adalah tampilan halaman edit daftar informasi yang dapat dilihat pada gambar 4.27.



Gambar 4.27 Tampilan Edit Informasi

4.1.1.24 Halaman Diagram Penjualan

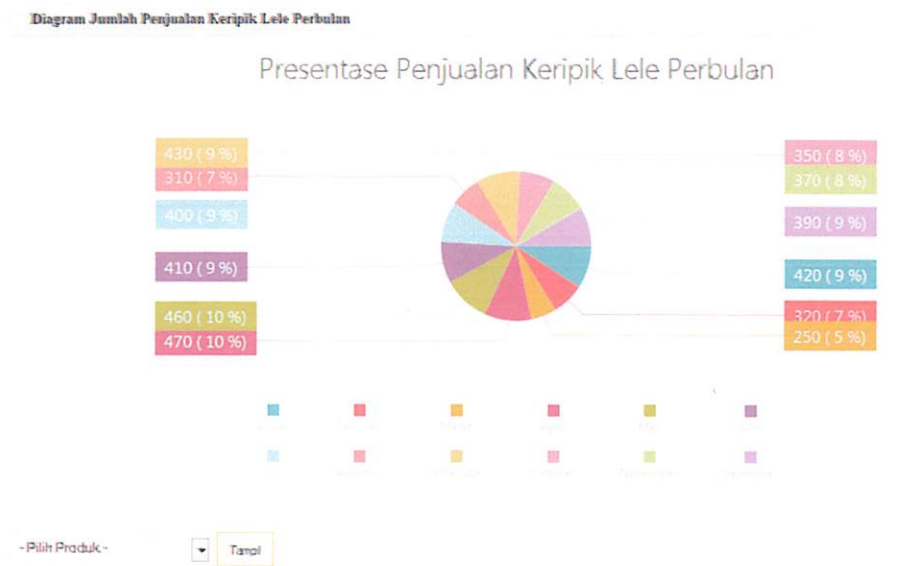
Halaman diagram penjualan merupakan halaman yang menampilkan diagram jumlah penjualan produk perbulan. Terdapat tiga jenis diagram produk yang dapat ditampilkan yaitu diagram produk keripik lele, nila dan wader. Berikut adalah tampilan diagram jumlah penjualan pada keripik lele yang dapat dilihat pada gambar 4.28.



Gambar 4.28. Tampilan Diagram Jumlah Penjualan Produk Keripik Lele Perbulan

4.1.1.25 Halaman Statistik Penjualan

Halaman statistik penjualan merupakan halaman yang menampilkan *chart* prosentase jumlah penjualan produk perbulan. Terdapat tiga jenis prosentase produk yang dapat ditampilkan yaitu pada produk keripik lele, nila dan wader Berikut adalah tampilan statistik prosentase penjualan keripik lele yang dapat dilihat pada gambar 4.29.



Gambar 4.29 Tampilan Statistik Prosentase Penjualan Produk Keripik Lele Perbulan

4.1.2 Bagian Pelanggan

Bagian pelanggan merupakan aplikasi berbasis *mobile* yang akan digunakan untuk para pelanggan dalam melakukan transaksi pemesanan produk. Pada bagian ini pelanggan hanya dapat melihat daftar informasi, daftar produk, histori yang di *managemen* oleh *administrator server*, melakukan pemesanan dan merubah profil data.

4.1.2.1 *Splash Screen*

Splash screen merupakan tampilan *loading* awal ketika aplikasi *mobile* dijalankan. Terdapat 2 fungsi *splash* yang ditampilkan yaitu *splash in*, ketika aplikasi pertama kali dibuka dan *splash out*, ketika aplikasi ditutup. Berikut tampilan *splash screen* yang dapat dilihat pada gambar 4.30.



Gambar 4.30 Tampilan Splash Screen

Gambar 4.31a merupakan tampilan dari *splash in*, gambar 4.31b merupakan tampilan dari *splash out*.

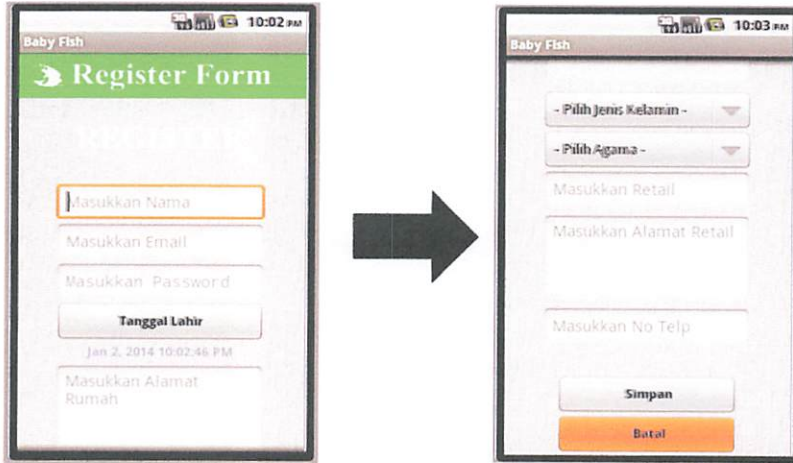
4.1.2.2 Form Login

Form login merupakan *form* yang digunakan untuk masuk ke dalam sistem. Terdapat *edit text email* dan *password* yang harus diisi oleh pelanggan. Jika *email* dan *password* benar, maka pelanggan akan masuk ke dalam sistem. Berikut adalah tampilan *form login* yang dapat dilihat pada gambar 4.31.

Gambar 4.31 Tampilan Form Login

4.1.2.3 Form Daftar Pelanggan

Form daftar pelanggan merupakan form yang digunakan apabila user belum terdaftar sebagai pelanggan dalam sistem. Pada form ini user diminta untuk mengisi data-data pribadi. Berikut adalah tampilan form daftar pelanggan yang dapat dilihat pada gambar 4.32.



Gambar 4.32 Tampilan Form Daftar Pelanggan

4.1.2.4 Menu Beranda

Menu beranda merupakan menu pertama kali yang ditampilkan pada saat pelanggan berhasil masuk ke dalam sistem. Pada menu ini menampilkan berbagai informasi pelanggan seperti daftar promo dan daftar produk baru yang akan dikeluarkan. Berikut adalah tampilan menu beranda yang dapat dilihat pada gambar 4.33.



Gambar 4.33 Tampilan Menu Beranda

Figure 1.1.1.1

... ..
... ..
... ..
... ..

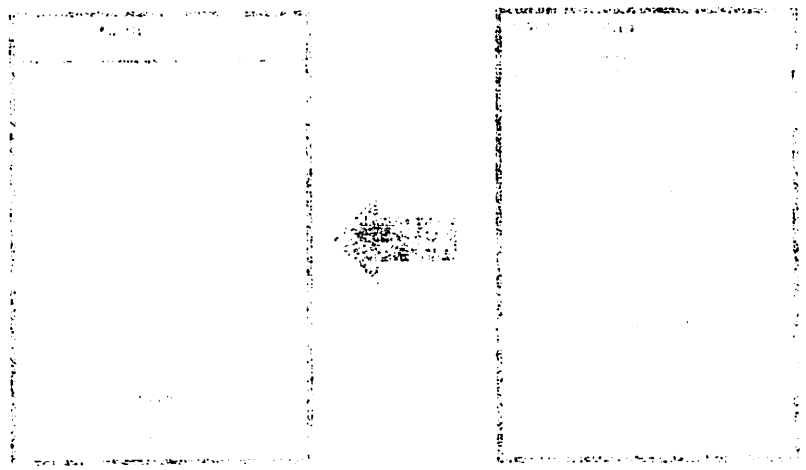


Figure 1.1.1.1

Figure 1.1.1.2

... ..
... ..
... ..
... ..



Figure 1.1.1.2

4.1.2.5 Menu Produk

Menu produk merupakan menu yang menampilkan daftar produk yang tersedia pada sistem administrasi. Berikut adalah tampilan menu produk yang dapat dilihat pada gambar 4.34.



Gambar 4.34 Tampilan Menu Produk

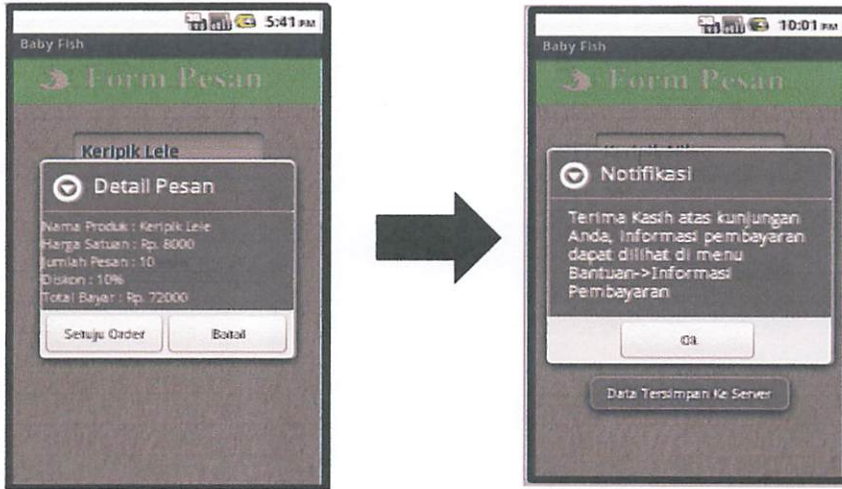
4.1.2.6 Form Pemesanan

Form pemesanan merupakan form untuk melakukan pemesanan produk. Dalam form ini pelanggan cukup menginputkan *quantity* produk ke dalam *edit text*. Berikut adalah tampilan form pemesanan yang dapat dilihat pada gambar 4.35.



Gambar 4.35 Tampilan Form Pemesanan

Ketika pelanggan menekan *button* pesan, maka akan muncul dialog detail pemesanan, apabila pelanggan setuju melakukan *order*, detail pemesanan akan dikirim ke *server* dan akan muncul dialog informasi pembayaran pada aplikasi *mobile*. Berikut adalah tampilan dialog detail pemesanan dan informasi pembayaran yang dapat dilihat pada gambar 4.36.



Gambar 4.36 Tampilan Dialog Detail Pemesanan dan Informasi Pembayaran

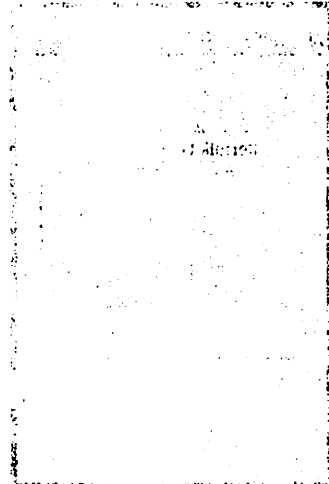
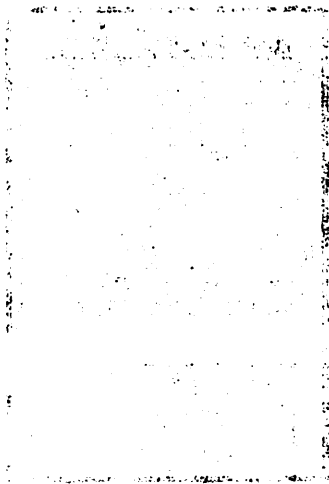
4.1.2.7 Menu Histori

Menu histori merupakan menu yang menampilkan daftar riwayat pemesanan produk yang pernah dilakukan pelanggan. Berikut adalah tampilan menu histori yang dapat dilihat pada gambar 4.37.



Gambar 4.37 Tampilan Menu Histori

This is a very important document which contains information regarding the
 activities of the organization in the past few years. It is a very
 confidential document and should be kept in a safe place. It is
 not to be distributed to anyone outside the organization.



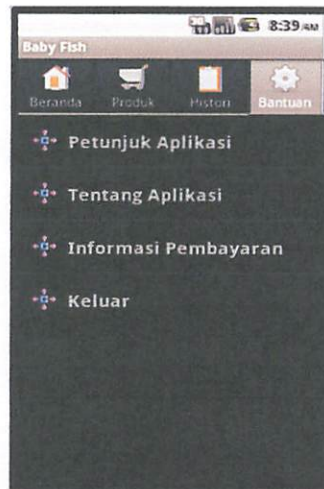
This is a very important document which contains information regarding the
 activities of the organization in the past few years. It is a very
 confidential document and should be kept in a safe place. It is
 not to be distributed to anyone outside the organization.



This is a very important document which contains information regarding the
 activities of the organization in the past few years. It is a very
 confidential document and should be kept in a safe place. It is
 not to be distributed to anyone outside the organization.

4.1.2.8 Menu Bantuan

Menu bantuan merupakan menu yang berisi informasi-informasi detail tentang aplikasi. Terdapat 4 *activity* dalam menu ini yaitu Petunjuk Aplikasi, Tentang Aplikasi, Informasi Pembayaran dan Keluar. Berikut adalah tampilan menu bantuan yang dapat dilihat pada gambar 4.38.



Gambar 4.38 Tampilan Menu Bantuan

Pada *activity* Petunjuk Aplikasi berisi tentang definisi dari menu-menu yang ada pada aplikasi *mobile*. Berikut adalah tampilan petunjuk aplikasi yang dapat dilihat pada gambar 4.39.

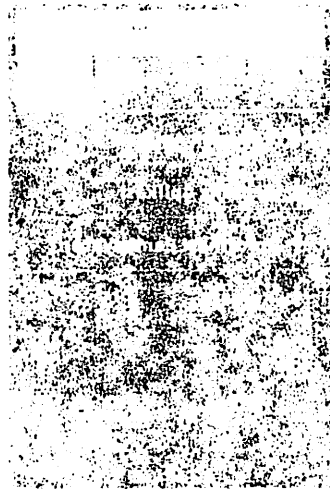


Gambar 4.39 Tampilan Petunjuk Aplikasi

1914-1915

... ..
... ..
... ..

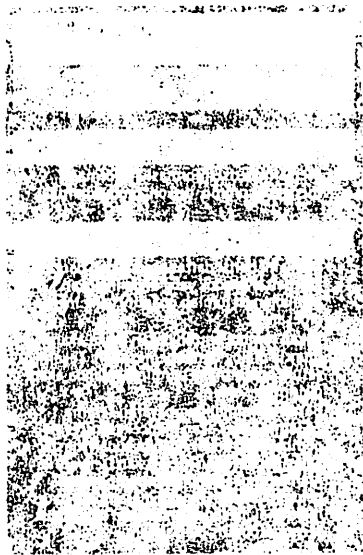
...



...

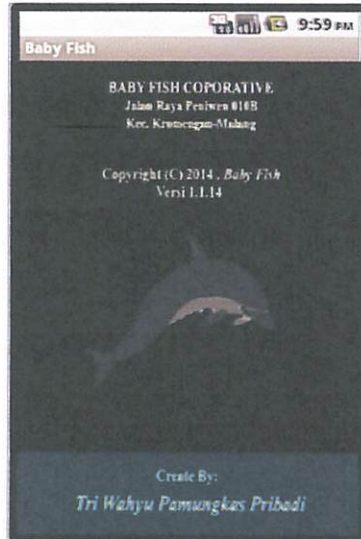
... ..
... ..

...



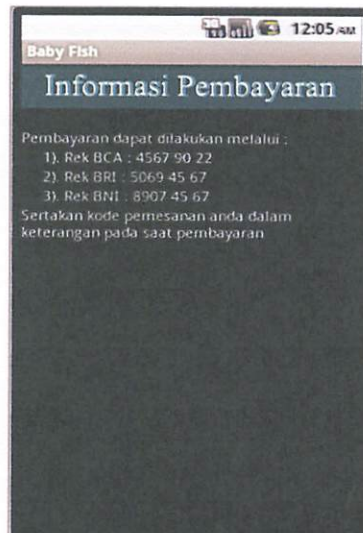
...

Pada *activity* Tentang Aplikasi berisi tentang informasi alamat *Owner Baby Fish*, versi aplikasi dan pembuat aplikasi. Berikut adalah tampilan tentang aplikasi yang dapat dilihat pada gambar 4.40.



Gambar 4.40 Tampilan Tentang Aplikasi

Pada *activity* Informasi Pembayaran berisi tentang petunjuk untuk melakukan proses pembayaran yang harus dilakukan pelanggan. Berikut adalah tampilan Informasi Pembayaran yang dapat dilihat pada gambar 4.41.

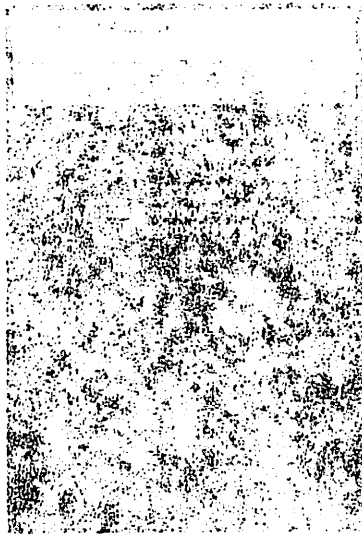


Gambar 4.41 Tampilan Informasi Pembayaran

adulterium in the same manner as in the case of adulteration of wine with water
and the adulteration of wine with water is a crime under the law of the State of
Ohio and the adulteration of wine with water is a crime under the law of the State of

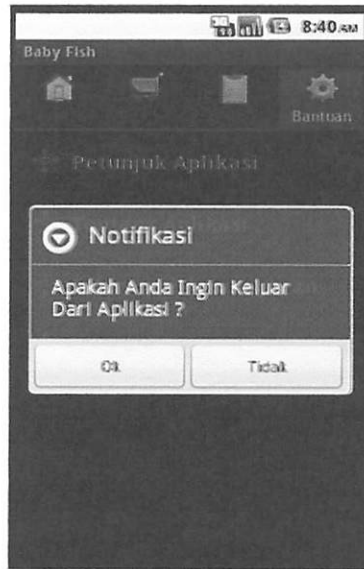


adulterium in the same manner as in the case of adulteration of wine with water
and the adulteration of wine with water is a crime under the law of the State of
Ohio and the adulteration of wine with water is a crime under the law of the State of



adulterium in the same manner as in the case of adulteration of wine with water

Pada *activity* Keluar merupakan pilihan untuk keluar dari aplikasi. Berikut adalah tampilan dialog keluar aplikasi yang dapat dilihat pada gambar 4.42.



Gambar 4.42 Tampilan Dialog Keluar Aplikasi

4.2 Pengujian

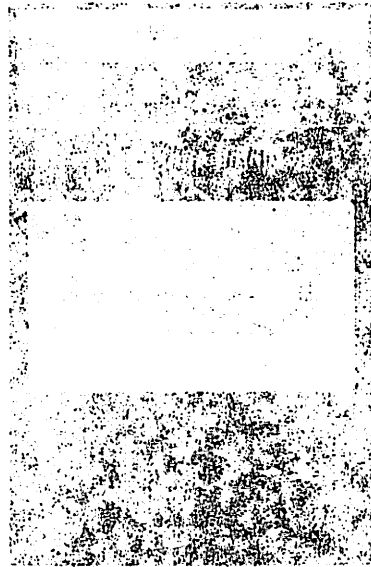
Pengujian adalah tahapan uji coba sistem dari aplikasi *web* dan *mobile* untuk mengetahui hasil dari implementasi yang telah dilakukan.

4.2.1 Pengujian Sistem

Pada bagian pengujian sistem yang dilakukan adalah pengujian terhadap fungsi dari aplikasi berupa proses penambahan data, perubahan data, penghapusan data, pemesanan produk dan koneksi *server* dan *client*. Pada tahapan ini dilakukan dengan dua proses pengujian, yaitu pengujian pada bagian aplikasi berbasis *web* dan pengujian pada bagian aplikasi *mobile*. Berikut adalah data-data yang diperoleh dari hasil pengujian sistem dengan menggunakan *web browser* Mozilla Firefox Versi 28.0 untuk aplikasi *web* dan Samsung Galaxy Mini 2 GT-S6500D dengan ukuran pixel 480x800 pada aplikasi *mobile* yang dapat dilihat pada tabel 4.1.

... ..

... ..



... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

... ..

Tabel 4.1 Hasil Pengujian Sistem Pada Aplikasi Web

Menu	Proses	Hasil	Prosentase
Login	Proses login	Y	100%
Pelanggan	Tambah data pelanggan	Y	100%
	Edit data Pelanggan	Y	100%
	Hapus data pelanggan	Y	100%
Ulang Tahun	Edit Ulang Tahun	Y	100%
	Hapus Ulang Tahun	Y	100%
Stok retail	Edit data Stok Retail	Y	100%
	Hapus data Stok Retail	Y	100%
Karyawan	Tambah data karyawan	Y	100%
	Edit data karyawan	Y	100%
	Hapus data karyawan	Y	100%
Produk	Tambah data produk	Y	100%
	Edit data produk	Y	100%
	Hapus data produk	Y	100%
Pesan	Edit status pesan	Y	100%
	Hapus data pesan	Y	100%
Penjualan	Tambah data penjualan	Y	100%
	Edit data penjualan	Y	100%
	Hapus data penjualan	Y	100%
Informasi	Tambah data informasi	Y	100%
	Edit data informasi	Y	100%
	Hapus data informasi	Y	100%
Kirim SMS	Tambah data sms	Y	100%
	Edit data sms	Y	100%
	Hapus data sms	Y	100%
Logout	Proses logout	Y	100%

Keterangan :

Y : berhasil

X : gagal

Berdasarkan hasil pengujian pada sistem yang diperoleh, semua fungsi pada aplikasi *web* dapat berjalan 100% dengan menggunakan *web browser* Mozilla Firefox. Untuk pengujian sistem pada aplikasi mobile dapat dilihat pada tabel 4.2.

Tabel 4.2 Hasil Pengujian Sistem Pada Aplikasi Mobile

Menu	Proses	Hasil	Prosentase
Register	Proses daftar pelanggan	Y	100%
Login	Proses login	Y	100%
Beranda	Tampilan Info dan Promo	Y	100%
Produk	Pesan Produk	Y	100%
Histori	Tampilan histori pesanan	Y	100%
Bantuan	Petunjuk Aplikasi	Y	100%
Logout	Proses logout	Y	100%

Keterangan :

Y : berhasil

X : gagal

Berdasarkan hasil pengujian pada sistem yang diperoleh, semua fungsi pada aplikasi *mobile* dapat berjalan 100% dengan menggunakan Samsung Galaxy Mini 2 GT-S6500D dengan ukuran pixel 480x800.

4.2.2 Pengujian Sistem Aplikasi Web Pada Beberapa Web Browser

Pada tahapan pengujian aplikasi *web* dilakukan dengan menggunakan beberapa *web browser*, dimana pengujian ini dilakukan untuk mengetahui tingkat keberhasilan dalam pembuatan aplikasi web. Berikut adalah tabel hasil pengujian aplikasi dengan beberapa *web browser* yang dapat dilihat pada tabel 4.3

Tabel 4.3 Hasil Pengujian Sistem Aplikasi Web dengan Beberapa Web Browser

No.	Proses	Browser					
		Mozila V. 28.0	Chrome V. 33.0	Opera V. 18.0	Safari V. 5.34	IE V.8	Maxthon V. 4.2.1
1	Tampilan aplikasi	Y	Y	Y	Y	T	Y
2	Proses login	Y	Y	Y	Y	Y	Y
3	Tambah data pelanggan	Y	Y	Y	Y	Y	Y
4	Edit data pelanggan	Y	Y	Y	Y	Y	Y
5	Hapus data pelanggan	Y	Y	Y	Y	Y	Y
6	Edit ulang tahun	Y	Y	Y	Y	Y	Y
7	Hapus ulang tahun	Y	Y	Y	Y	Y	Y
8	Edit data stok retail	Y	Y	Y	Y	Y	Y

9	Hapus data stok retail	Y	Y	Y	Y	Y	Y
10	Tambah data karyawan	Y	Y	Y	Y	Y	Y
11	Edit data karyawan	Y	Y	Y	Y	Y	Y
12	Hapus data karyawan	Y	Y	Y	Y	Y	Y
13	Tambah data produk	Y	Y	Y	Y	Y	Y
14	Edit data produk	Y	Y	Y	Y	Y	Y
15	Hapus data produk	Y	Y	Y	Y	Y	Y
16	Edit status pesan	Y	Y	Y	Y	Y	Y
17	Hapus data pesan	Y	Y	Y	Y	Y	Y
18	Tambah data penjualan	Y	Y	Y	Y	Y	Y
19	Edit data penjualan	Y	Y	Y	Y	Y	Y
20	Hapus data penjualan	Y	Y	Y	Y	Y	Y
21	Tambah data informasi	Y	Y	Y	Y	Y	Y
22	Edit data informasi	Y	Y	Y	Y	Y	Y
23	Hapus data informasi	Y	Y	Y	Y	Y	Y
24	Tambah data sms	Y	Y	Y	Y	Y	Y
25	Edit data sms	Y	Y	Y	Y	Y	Y
26	Hapus data sms	Y	Y	Y	Y	Y	Y

Keterangan :

Y : berhasil

X : gagal

Pada hasil pengujian aplikasi dengan menggunakan 6 *web browser* yang berbeda, semua tampilan dan fungsi aplikasi pada aplikasi *web* dapat berjalan 100% pada *web browser* Mozilla Firefox, Chrome, Safari, Maxthon, Opera dan 96,65% pada *web browser* Internet Explorer.

4.2.3 Pengujian Sistem Aplikasi *Mobile* Pada Beberapa Telepon Genggam

Pada tahapan ini pengujian aplikasi *mobile* dilakukan pada beberapa telepon genggam dengan berbagai versi dari *android*. Dimana pengujian ini dilakukan untuk mengetahui tingkat keberhasilan fungsional dalam pembuatan aplikasi. Berikut adalah tabel hasil pengujian fungsional pada aplikasi *mobile* pada beberapa telepon genggam yang dapat dilihat pada tabel 4.4.

Tabel 4.4 Hasil Pengujian Sistem Aplikasi Mobile Dengan Beberapa Type dan Pixel

No.	Type	Pixel	Posisi Telepon			
			Landscape		Potrait	
			H	P	H	P
1	Samsung Galaxy Y S5360	240 x 320	Y	100%	Y	100%
2	Samsung Galaxy Mini 2 S6500D	480 x 800	Y	100%	Y	100%
3	Samsung Galaxy S4 GT-19500	540 x 960	Y	0%	Y	0%
4	Smartfren Andromax U2.V1.00	540 x 960	Y	0%	Y	0%
5	Idea Tab A1000-G	1024 x 600	Y	0%	Y	0%

Keterangan :

Y : berhasil

H : Hasil

X : gagal

P : Prosentase

Pada hasil pengujian sistem aplikasi *mobile* dengan menggunakan beberapa telepon genggam dengan berbagai type dan ukuran pixel yang berbeda yaitu 240 x 320, 480 x 800, 540 x 960 dan 1024 x 600 aplikasi dapat berjalan 40% baik dengan posisi *landscape* maupun *potrait*.

Tabel 4.5 Hasil Pengujian Sistem Aplikasi Mobile Dengan Beberapa Sistem Operasi

No.	Android Operating System	Versi	Hasil	Prosentase
1	Froyo Yoghurt (Froyo)	2.2.1	Y	100%
2	Ginger Bread	2.3.3	Y	100%
3	Ice Cream Sandwich	4.0.0	Y	0%
4	Jelly Bean	4.1.2	Y	0%

Keterangan :

Y : berhasil

X : gagal

Pada tabel 4.5 menjelaskan bahwa aplikasi *mobile* ini dapat berjalan 50%. pada *operating sistem* yang diuji antara versi Froyo (2.2.1) hingga versi Jelly Bean (4.1.2).

4.2.4 Pengujian Proses *Sales Force Automation*

Pada bagian pengujian proses *Sales Force Automation* ini yang dilakukan adalah melakukan pengujian terhadap fitur-fitur *SFA* yang ada pada sistem Aplikasi *Sales Force Automation* Berbasis *Android*. Pada tahap ini dilakukan proses dua pengujian, yaitu pengujian terhadap operasi *Sales Force Automation* dan pengujian *Auto Notification* terhadap pelanggan. Berikut hasil pengujian operasi *Sales Force Automation* yang dapat dilihat pada tabel 4.6.

Tabel 4.6 Hasil Pengujian Operasi *Sales Force Automation* Pada Sistem

No	Proses Otomasi Sistem	Hasil	Prosentase
1	<i>Update</i> jumlah stok keripik berkurang sesuai jumlah permintaan ketika status pemesanan dirubah menjadi "Ter kirim"	<i>Y</i>	100%
2	<i>Update</i> jumlah stok keripik menjadi jumlah stok awal dalam jangka satu hari	<i>Y</i>	100%
3	<i>Update</i> jumlah penjualan keripik perbulan ketika status pemesanan dirubah menjadi "Ter kirim"	<i>Y</i>	100%
4	Pemberian diskon otomatis berdasarkan histori pembelian retail	<i>X</i>	0%
5	Informasi promo, diskon, produk baru pada aplikasi <i>mobile</i>	<i>Y</i>	100%
6	Menampilkan grafik jumlah penjualan pertahun	<i>Y</i>	100%

Keterangan :

Y : berhasil

X : gagal

Pada tabel 4.6 menjelaskan bahwa pengujian proses otomasi sistem yang dilakukan dapat berjalan 83,33%. Selanjutnya hasil pengujian *Auto Notification* pada retail dapat dilihat pada tabel 4.7.

Tabel 4.7 Hasil Pengujian Auto Notification dan Manual Notification Pada Retail

No	Proses Auto Notification	Hasil	Prosentase
1	Notifikasi <i>Auto Respon</i> pada pendaftaran pelanggan baru	Y	100%
2	Notifikasi <i>Auto Respon</i> pada pelanggan mengenai informasi pembayaran	X	0%
3	Ucapan selamat ulang tahun kepada pelanggan	Y	50%
4	Ucapan selamat hari raya kepada pelanggan	Y	50%
5	Notifikasi mengenai produk baru, promosi dan diskon kepada pelanggan	Y	50%

Keterangan :

Y : berhasil

X : gagal

Pada tabel 4.7 menjelaskan bahwa pengujian *Auto Notification* dan *Manual Notification* kepada pelanggan dapat berjalan 50% yang mana pada pengujian ini dilakukan dengan 2 nomer provider yang berbeda.

Pada tabel 4.8 menjelaskan bahwa hasil dari pengujian durasi *Auto Notification* dan *Manual Notification* dengan menggunakan 2 jenis provider yang berbeda dan dilakukan sebanyak 3 kali pengujian. Berikut hasil pengujian durasi yang dapat dilihat pada tabel 4.8.

Tabel 4.8 Hasil Pengujian Durasi Auto Notification dan Manual Notification

No	Proses Auto Notification	Provider					
		IM3			XL		
		P1	P2	P3	P1	P2	P3
1	Notifikasi <i>Auto Respon</i> pada pendaftaran pelanggan baru	30s	40s	32s	15s	12s	14s
		Rata-rata: 34s			Rata-rata: 13,67s		
2	Notifikasi <i>Auto Respon</i> pada pelanggan mengenai informasi pembayaran	x	x	x	x	x	X
		Rata-rata:x			Rata-rata:x		
3	Ucapan selamat ulang tahun kepada pelanggan	34s	32s	33s	10s	13s	10s
		Rata-rata: 33s			Rata-rata: 11s		
4	Ucapan selamat hari raya kepada pelanggan	27s	28s	28s	18s	12s	15s
		Rata-rata: 27,67s			Rata-rata: 15s		
5	Notifikasi mengenai produk baru,	31s	22s	43s	12s	13s	12s

	promosi dan diskon kepada pelanggan	Rata-rata:32s	Rata-rata:12,33s
--	-------------------------------------	---------------	------------------

Keterangan :

Y : berhasil

X : gagal

S: Second (Satuan waktu dala detik)

Pada tabel 4.8 dijelaskan bahwa pada pengujian notifikasi *Auto Respon* pada pendaftaran pelanggan baru yang dilakukan diperoleh hasil durasi dengan nilai rata-rata 34s pada *provider* IM3 dan 13,67s pada *provider* XL. Pada pengujian notifikasi *Auto Respon* pada pelanggan mengenai informasi pembayaran diperoleh hasil durasi dengan nilai rata-rata 0 pada *provider* IM3 dan 0 pada *provider* XL. Pada pengujian ucapan selamat ulang tahun kepada pelanggan diperoleh hasil durasi dengan nilai rata-rata 33s pada *provider* IM3 dan 11s pada *provider* XL. Pada pengujian ucapan selamat hari raya kepada pelanggan diperoleh hasil durasi dengan nilai rata-rata 27,67s pada *provider* IM3 dan 15s pada *provider* XL. Pada pengujian notifikasi mengenai produk baru, promosi dan diskon kepada pelanggan diperoleh hasil durasi dengan nilai rata-rata 32s pada *provider* IM3 dan 12,33s pada *provider* XL.

Dari hasil 5 pengujian proses *auto notification* dengan 2 *provider* yang berbeda, diperoleh hasil dengan nilai rata-rata 25,34s pada *provider* IM3 dan 10,4s pada *provider* XL.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melalui beberapa tahapan perancangan, implementasi dan pengujian aplikasi *Sales Force Automation (SFA)* pada sistem penjualan keripik *Baby Fish* berbasis *Android*, maka dapat diambil kesimpulan bahwa:

1. Pada Aplikasi web, dari segi fungsional aplikasi dapat berjalan 100% dan pada pengujian 6 web browser, aplikasi dapat berjalan 100% pada 5 *web browser*, yaitu Mozilla Firefox V.28.0, Opera V.18.0, Maxthon V.4.2.1, Chrome V.33.0 dan Safari V.5.34, 95,65% pada *web browser* Internet Explorer V.8.
2. Pada aplikasi *mobile*, dari segi fungsional aplikasi dapat berjalan 100% yang diuji menggunakan Samsung Galaxi Mini 2, pada pengujian dengan beberapa type dan pixel, aplikasi dapat berjalan 40% baik dengan posisi *landscape* maupun *potrait* dan pada pengujian berbagai jenis *operating system*, aplikasi dapat berjalan 50%. pada *operating sistem* yang diuji antara versi Froyo (2.2.1) hingga versi Jelly Bean (4.1.2).
3. Pada hasil pengujian proses *auto notification* dan *manual notification* dengan 2 *provider* yang berbeda yaitu IM3 dan XL, diperoleh hasil durasi dengan nilai rata-rata 25,34s pada *provider* IM3 dan 10,04s pada *provider* XL.
4. Proses otomasi stok keripik pada retail pada saat retail melakukan pemesanan dan status pemesanan diubah menjadi “Ter kirim”, maka secara otomatis stok keripik akan berkurang sesuai dengan permintaan dan dalam jangka satu hari stok keripik akan kembali pada jumlah semula. Hal ini sangat membantu *admin*, sehingga *admin* tidak perlu melakukan *update manual* pada saat terjadi pemesanan keripik.
5. Penerapan *Sales Force Automation* mempermudah *admin* dalam memonitoring jumlah stok keripik yang masih tersedia, sehingga ketika retail melakukan permintaan produk, stok akan selalu terpenuhi.

6. Proses *update* jumlah penjualan perbulan secara otomatis ketika status pemesanan dirubah menjadi “Ter kirim”, sehingga mempermudah *admin* dalam menganalisa jumlah penjualan keripik perbulan dan pertahun.

5.2 Saran

Aplikasi *Sales Force Automation (SFA)* pada sistem penjualan keripik *Baby Fish* berbasis *Android* ini dapat dikembangkan. Adapaun saran-saran yang dapat dilakukan terhadap pengembangan aplikasi ini adalah sebagai berikut:

1. Membuat struk pemesanan secara otomatis ketika status pesanan retail dirubah menjadi “Ter kirim”.
2. Menambahkan *widget combobox* pertahun pada menu diagram dan statistik penjualan sehingga dapat menampilkan diagram dan statistik penjualan berdasarkan tahun yang dipilih sehingga fungsi menu menjadi lebih dinamis.
3. Menambahkan menu *report* penjualan dan dapat ditampilkan berdasar kategori penjualan produk, penjualan berdasarkan id pelanggan, penjualan bulan ini dan penjualan tahun ini.

DAFTAR PUSTAKA

- [1] Arief, M Rudyanto. 2011. *Pemrograman Web Dinamis Menggunakan PHP & MySQL*. Andi Yogyakarta.
- [2] Kadir, Abdul. 2008. *Dasar Pemrograman Web Dinamis Menggunakan PHP*. Andi. Yogyakarta.
- [3] Amin, Rifan. 2013. *Perancangan Aplikasi Mobile Pemesanan Menu Makanan Pada Rumah Makan Carita Berbasis Android*. Institut Teknologi Nasional. Malang.
- [4] Irawan, Yudie. 2011. *Perancangan Sistem Informasi Perpustakaan Berbasis Web Application*. Universitas Diponegoro. Semarang.
- [5] Sugiarto. 2013. *Panduan Wisata Kabupaten Malang Berbasis Aplikasi Mobile Phone*. Institut Teknologi Nasional. Malang.
- [6] Andriyanto, Bagus, Rahmad; Mudjahidin; Sani, Asrul, Nifsu. 2011. *Pembuatan Sales Force Automation (SFA) Berbasis Android Dengan Menggunakan Eclipse ADT Plugin Studi Kasus Pada Supplier XYZ*. Institut Teknolgi Sepuluh November. Surabaya.
- [7] Akbar, Januar, Rizki; Sarno, Riyanarto; Sukmandityo, Rino. 2011. *Implementasi Customer Relationship Management (CRM) Sales Force Automation Dan Reporting Menggunakan Metode Service Oriented Architecture (SOA)*. Institut Teknologi Sepuluh November. Surabaya.
- [8] Rusda, Arini; Sarno, Riyanarto; Sunaryono, Dwi. 2011. *Pembangunan Aplikasi Customer Relationship Management Berorientasi Service Pada Platform Java Menggunakan Framework Apache Open For Bussiness*. Institut Teknologi Sepuluh November. Surabaya.
- [9] Falahah; Nuraini, Alfina; Yunarso, Widhi, Eka. 2011. *Analisis Dan Perancangan E-Service Berbasis Customer Relationship Management Dengan Segmentasi Rumah Tangga Di PT. PLN UP Bojonegoro*. Politeknik Telkom. Bandung.
- [10] Geni, Lenggo. 2011. *Perancangan Website Untuk Pengambil Keputusan Pelanggan Pada PT. BESTINDO AQUATEK SEJAHTERA Dengan Menggunakan PHP, MySQL Dan Ajax*. Universitas Gunadarma.



**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo KM.2, Malang**

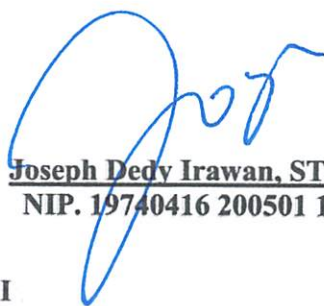
**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Tri Wahyu Pamungkas Pribadi
NIM : 10.18.018
Program Studi : Teknik Informatika S1
Judul : Implementasi *Sales Force Automation* Berbasis *Android* Pada
Usaha Keripik *Baby Fish*.

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada:

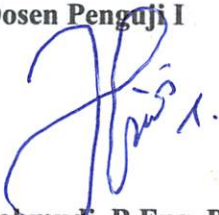
Hari : Selasa
Tanggal : 18 Februari 2014
Nilai : 87.67 (A)

**Panitia Penguji Skripsi,
Ketua Majelis Penguji**



Joseph Dedy Irawan, ST, MT.
NIP. 19740416 200501 1 002

Dosen Penguji I



Ali Mahmudi, B.Eng, PhD
NIP. P. 103 100 0 429

Dosen Penguji II



Michael Ardita, ST. MT.
NIP. P. 103 100 0 434



BERITA ACARA UJIAN PRAKTIK
FAKULTAS TEKNOLOGI INDUSTRI

Nama : ... NIM : ... Program Studi : ... Judul : ...	Nama : ... NIM : ... Program Studi : ... Judul : ...
---	---

Dipertanggungjawabkan oleh Ketua Program Studi dan Sekretaris pada tanggal ...
 Hari : ...
 Tanggal : ...
 Waktu : ...

Ketua Program Studi
 Sekretaris Program Studi

(Signature)
 Nama dan NIM

(Signature)
 Nama dan NIM

(Signature)
 Nama dan NIM

(Signature)
 Nama dan NIM

(Signature)
 Nama dan NIM



**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo KM.2, Malang**

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Program Studi Teknik Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa:

Nama : Tri Wahyu Pamungkas Pribadi
NIM : 10.18.018
Program Studi : Teknik Informatika S1
Judul : Implementasi *Sales Force Automation* Berbasis *Android* Pada Usaha Keripik *Baby Fish*.

No.	Penguji	Tanggal	Uraian	Paraf
1	Penguji I	18 Februari 2014	<ul style="list-style-type: none">- Pengujian browser tuliskan browser versi berapa.- Tambahkan status pesanan (Proses, Batal, Lunas, Terkirim).- Hal 61 spasi 1,5- Tabel 4.7 hal 66. Kalau pernah berhasil dan sekarang tidak, tulis hasil 50%.- Tabel 4.8 hal 66. Pada 4.7 no 2 gagal, pada 4.8 no 2 mestinya gagal / gak ada data.- Lampirkan beberapa script.- Penerjelas gambar 3.6, 3.7, 3.12.	
2	Penguji II	18 Februari 2014	<ul style="list-style-type: none">- Tambahkan konfigurasi antara web server – sms dan android pada arsitektur sistem.	

			<ul style="list-style-type: none">- Tambahkan flowchart untuk proses notifikasi.- Hitungan pada kesimpulan perlu disesuaikan lagi dengan hasil pengujian.	
--	--	--	--	--

Dosen Penguji I

Ali Mahmudi, B.Eng, PhD
NIP. P. 103 100 0 429

Dosen Penguji II

Michael Ardita, ST. MT.
NIP. P. 103 100 0 434

Mengetahui

Dosen Pembimbing I

Joseph Dedy Irawan, ST, MT.
NIP. 19740416 200501 1 002

Dosen Pembimbing II

Suryo Adi Wibowo, ST, MT.
NIP.P. 1031000438



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 21 Oktober 2013

Nomor : ITN-70/INF/TA/2013
Lampiran : ---
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu **Joseph Dedy Irawan, ST, MT**
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : TRI WAHYU PAMUNGKAS PRIBADI
Nim : 1018018
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 OKTOBER 2013 S/D 21 MARET 2014

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,



Joseph Dedy Irawan, ST., MT.
NIP : 197404162005021002

Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 21 Oktober 2013

Nomor : ITN-70/INF/TA/2013
Lampiran : ---
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu **Suryo Adi Wibowo, ST.MT**
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : TRI WAHYU PAMUNGKAS PRIBADI
Nim : 1018018
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

21 OKTOBER 2013 S/D 21 MARET 2014

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,

Joseph Dedy Irawan, ST., MT.
NIP : 197404162005021002

Form S-4a



**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo KM.2, Malang**

FORMULIR BIMBINGAN SKRIPSI

Nama : Tri Wahyu Pamungkas Pribadi
NIM : 1018018
Masa Bimbingan : 21 Oktober 2013 – 21 Maret 2014
Judul Skripsi : Implementasi *Sales Force Automation* Berbasis *Android* Pada Usaha Keripik *Baby Fish*

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	27-11-2013	Bimbingan Laporan Bab 1-3	
2	16-12-2013	Revisi Laporan Bab 1-3	
3	17-12-2013	Demo Program Secara Offline	
4	16-01-2014	Bimbingan Laporan Bab 4-5	
5	20-01-2014	Bimbingan Jurnal Seminar Hasil	
6	11-02-2014	Bimbingan Laporan Bab 1-5	

Malang, 13 Februari 2014
Dosen Pembimbing

Joseph Dedy Irawan, ST., MT.
NIP : 197404162005011002



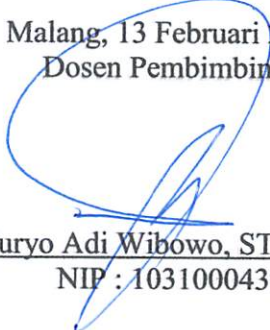
**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
Jl. Karanglo KM.2, Malang**

FORMULIR BIMBINGAN SKRIPSI

Nama : Tri Wahyu Pamungkas Pribadi
NIM : 1018018
Masa Bimbingan : 21 Oktober 2013 – 21 Maret 2014
Judul Skripsi : Implementasi *Sales Force Automation* Berbasis *Android* Pada Usaha Keripik *Baby Fish*

No.	TANGGAL	URAIAN	PARAF PEMBIMBING
1	20-11-2013	Bimbingan Laporan Bab 1-3	
2	22-11-2013	Revisi Laporan Bab 1-3	
3	28-11-2013	Revisi Laporan Bab 1-3	
4	19-12-2013	Demo Program Secara Offline	
5	15-01-2014	Bimbingan Laporan Bab 4-5	
6	16-01-2014	Revisi Laporan Bab 4-5	
5	16-01-2014	Bimbingan Jurnal Seminar Hasil	
6	17-01-2014	Revisi Jurnal Seminar Hasil	
7	18-01-2014	Revisi Jurnal Seminar Hasil	
8	11-02-2014	Bimbingan Laporan Bab 1-5	

Malang, 13 Februari 2014
Dosen Pembimbing


Suryo Adi Wibowo, ST., MT.
NIP : 1031000438

Source Code

Aplikasi Web

set_database.php

```
<?php

class database {
    private $db_host;
    private $db_user;
    private $db_pass;
    private $db_base;

    public function __construct($d_host, $d_user,
                                $d_pass, $d_base) {
        $this->db_host = $d_host;
        $this->db_user = $d_user;
        $this->db_pass = $d_pass;
        $this->db_base = $d_base;
    }

    public function koneksi_mysql() {
        mysql_connect($this->db_host, $this->db_user,
                    $this->db_pass);
        mysql_select_db($this->db_base);
    }
}

?>
```

set_koneksi.php

```
<?php

require ('set_database.php');

$server = 'localhost';
$username = 'root';
$password = '';
$database = 'babyfish_database';

?>
```

set_init.php

```
<?php

session_start();

require_once ('set_koneksi.php');
require_once ('set_login.php');
require_once ('set_halaman.php');

$db = new database($server, $username, $password, $database);
$log = new login_setting();
$db->koneksi_mysql();

?>
```

set_login.php

```
<?php

class login_setting {

    public function panggil_css () {
        echo "<html><head><link rel='stylesheet' type='text/css'
            href='../Konfigurasi/css/login.css'>";
    }

    public function validasi () {
        echo "<linkhref='../Konfigurasi/css/
            validationEngine.jquery.css' rel='stylesheet'>
            <script src='../Konfigurasi/js/validasi/
            jquery-1.8.2.min.js'></script>
            <script src='../Konfigurasi/js/validasi/
            jquery.validationEngine-en.js'></script>
            <script src='../Konfigurasi/js/validasi/
            jquery.validationEngine.js'></script>
            <script>
                $(document).ready(function() {
                    $('#form').validationEngine();
                });
            </script></head>";
    }

    public function header(){
        echo "<body><header id='header'>
            <div class='lebarHeader'><hgroup id='hgroup'>
            <section id='section'><img src='../
            Desain/logo.png'></section>
            <section id='judul'><h1>Aplikasi Sales Force
                Automation<br>Pada Penjualan Keripik Baby
                Fish</h1></section></div></hgroup></header>";
    }

    public function tampilan_form () {
        echo" <section id='container'><div class='login' id='form'>
            <h1>Silakan Login</h1>
            <table>
            <form method='POST' action='cek_user.php' >
            <tr><td><input type='text' name='username'
                placeholder='Masukkan Email' class='input-block-level
                validate[required,custom[email]]'></td></tr>
            <tr><td><input type='password' name='password'
                placeholder='Masukkan Password' class='input-block-
                level validate[required]'></td></tr>
            <tr><td><p class='submit'><input type='submit'
                name='submit' value='Login'></p></td></tr>
            </form></table></div></section>";
    }
}
```

```

public function footer () {
    echo "<footer id='footer'><h1>Create By : Tri Wahyu
        Pamungkas Pribadi<br>Institut Teknologi Nasional
        Malang<br>Informatika 2010</h1></footer>
        </body></html>";
}

public function cek_login_user () {
    $username = $_POST[username];
    $password = md5($_POST[password]);

    $username = mysql_escape_string($username);
    $password = mysql_escape_string($password);
    $que = mysql_query("SELECT * FROM _user WHERE email=
        '$username' AND password='$password' ");
    $log = mysql_fetch_array($que);
    $num = mysql_num_rows($que);

    if ($num > 0) {
        session_start();
        session_register("username");
        session_register("password");
        session_register("nama");

        $_SESSION[username] = $log[email];
        $_SESSION[password] = $log[password];
        $_SESSION[nama] = $log[nama];
        header('location:administrasi.php?module=beranda');
    } else { echo "hahaha"; }
}
}
?>

```

set_admin.php

```

<?php

class admin_setting {

public function koneksi_css () {
    $url = $_SERVER['REQUEST_URI'];
    header("Refresh: 10; URL=$url");

    echo "<html><head><title>Baby Fish Sistem</title>
        <link rel='stylesheet' type='text/css'
        href='../Konfigurasi/css/admin.css'>
        <link href='../Konfigurasi/css/
        validationEngine.jquery.css' rel='stylesheet'>";
}

public function validasi () {
    echo "<script src='../Konfigurasi/js/validasi/
        jquery-1.8.2.min.js'></script>
        <script src='../Konfigurasi/js/validasi/
        jquery.validationEngine-en.js'></script>";
}
}

```

```

<script src='../..//Konfigurasi/diagram/
penjualan/jquery-1.9.1.min.js'></script>
<script>
    $(document).ready(function() {
        $('#form').validationEngine();
    });
</script>
<script src='../..//Konfigurasi/js/validasi/
jquery.validationEngine.js'></script>
</head>";
}

public function timer() {
    if(!isset($_GET['cek'])) {
        include "set_timer.php";
    }
}

public function header () {
    echo "<body><header id='header'><hgroup id='hgroup'>
        <section id='logo'><img src = '../..//Desain/
            logo.png'></section>
        <div id='icon_admin'><img src = '../..//Desain/
            icon_admin.png'> $_SESSION[nama] <span> | </span>
            <img src = '../..//Desain/icon_keluar.png'><a href=
            'keluar.php'> Keluar</a></div></hgroup>";
}

public function isi_modul () {
    echo "<section id='bar'></section><aside id='menu'>
        <form class='pencarian' method='POST'
            action='?module=cari'>
            <input type='text' name='cari' placeholder='Pencarian
            Data' onfocus='if(!this._haschanged){this.value='';
            this._haschanged=true;''></form><hr>
        <ul>
<li class='beranda'><a href=?module=beranda>Beranda</a></li>
<li class='pelanggan'><a href=?module=pelanggan>Daftar
Pelanggan</a></li>
<li class='ultra'><a href=?module=ulangtahun>Daftar Ulang
Tahun</a></li>
<li class='stok'><a href=?module=stok_retail>Daftar Stok
Retail</a></li>
<li class='karyawan'><a href=?module=karyawan>Daftar
Karyawan</a></li>
<li class='produk'><a href=?module=produk>Daftar Produk</a></li>
<li class='pesan'><a href=?module=pesan>Daftar Pesanan</a></li>
<li class='penjualan'><a href=?module=penjualan>Daftar
Penjualan</a></li>
<li class='informasi'><a href=?module=informasi>Daftar
Informasi</a></li>
<li class='msgateway'><a href=?module=msgateway>Kirim
SMS</a></li>
<li class='diagram'><a href=?module=diagram>Diagram
Penjualan</a></li>
<li class='statistik'><a href=?module=prosentase>Prosentase

```

```

Penjualan</a></li>
        </ul></aside>";
    }

    public function load_modul() {
        include "set_modul.php";
        echo "</header></body></html>";
    }
}
?>

```

set_load.php

```

<?php
require ('../konfigurasi/set_admin.php');

class load_konten extends admin_setting
{
    public function load_konten_data ()
    {
        $get_konten = new admin_setting();
        $get_konten->koneksi_css();
        $get_konten->validasi();
        $get_konten->timer();
        $get_konten->header();
        $get_konten->isi_modul();
        $get_konten->load_modul();
    }
}
?>

```

set_module.php

```

<?php
include "../konfigurasi/set_init.php";

if($_GET[module]=='beranda')
{
    include "../aplikasi/Beranda/beranda.php";
    $get_module = new modul_beranda();
    $get_module -> crud_beranda();
}

elseif ($_GET[module]=='pelanggan')
{
    include "../aplikasi/DaftarPelanggan/_set_pelanggan.php";
    $get_module = new modul_pelanggan();
    $get_module -> crud_pelanggan();
}

elseif ($_GET[module]=='ulangtahun') {

```

```

        include "../aplikasi/DaftarUlangTahun/_set_ulta.php";
        $get_module = new modul_ulang_tahun();
        $get_module -> crud_ulang_tahun();
    }

elseif ($_GET[module]=='stok_retail') {
    include "../aplikasi/DaftarStokRetail/_set_stok.php";
    $get_module = new modul_stok_retail();
    $get_module -> crud_stok_retail();
}

elseif ($_GET[module]=='karyawan')
{
    include "../aplikasi/DaftarKaryawan/_set_karyawan.php";
    $get_module = new modul_karyawan();
    $get_module -> crud_karyawan();
}

elseif ($_GET[module]=='produk')
{
    include "../aplikasi/DaftarProduk/_set_produk.php";
    $get_module = new modul_produk();
    $get_module -> crud_produk();
}

elseif ($_GET[module]=='pesan')
{
    include "../aplikasi/DaftarPesan/_set_pesan.php";
    $get_module = new modul_pesan();
    $get_module -> crud_pesan();
}

elseif ($_GET[module]=='penjualan')
{
    include "../aplikasi/DaftarPenjualan/_set_penjualan.php";
    $get_module = new modul_penjualan();
    $get_module -> crud_penjualan();
}

elseif ($_GET[module]=='informasi')
{
    include "../aplikasi/InformasiPelanggan/
        _set_informasi.php";
    $get_module = new modul_informasi();
    $get_module-> crud_informasi();
}

elseif ($_GET[module]=='msggateway')
{
    include "../aplikasi/DaftarSms/_set_sms.php";
    $get_module = new modul_sms();
    $get_module-> crud_sms();
}

elseif ($_GET[module]=='diagram')
{

```

```

        include "../aplikasi/DiagramStatistik/
                diagram_penjualan.php";
        $get_module = new modul_diagram_penjualan();
        $get_module -> crud_diagram_penjualan();
    }

elseif ($_GET[module]=='prosentase')
{
    include "../aplikasi/DiagramStatistik/
            statistik_penjualan.php";
    $get_module = new modul_statistik_penjualan();
    $get_module -> crud_statistik_penjualan();
}

?>

```

login.php

```

<?php

require ('../konfigurasi/set_login.php');

$load_form = new login_setting();
$load_form->panggil_css();
$load_form->validasi();
$load_form->header();
$load_form->tampilan_form();
$load_form->footer();

?>

```

cekuser.php

```

<?php

require ('../konfigurasi/set_init.php');
$log->cek_login_user();

?>

```

administrasi.php

```

<?php
require ('set_load.php');

$get_link = new load_konten();

session_start();
if (empty($_SESSION[username]) AND empty($_SESSION[password]))
    echo "Ho";
else
    $get_link->load_konten_data();

?>

```

set_modul.php

```
<?php
class eksekusi_kontrol
{
    public $module;
    public $action;

    public function set_module ($x)
    {
        $this->module = $x;
    }

    public function set_action ($x)
    {
        $this->action = $x;
    }

    public function get_module ()
    {
        return $this->module;
    }

    public function get_action ()
    {
        return $this->action;
    }
}
?>
```

beranda.php

```
<?php
class modul_beranda {
    public function tampilan_selamat_datang() {
        echo "<section id='konten'>
            <h2 class='kotak'>Selamat Datang</h2>";
    }

    public function tampilan_informasi_baru() {
        echo "<h2 class='info'>Informasi Baru</h2>";

        $stampil = mysql_query("SELECT * FROM _informasi_pelanggan
ORDER BY id_informasi DESC LIMIT 3");
        while ($a=mysql_fetch_array($stampil)) {
            echo "<h4 class='isi_informasi'>
                <div id='spasi'>$a[judul]</div>
                <div id='spasi'> $a[deskripsi]</div>
                <div id='spasi_poajok'>$a[tanggal]</div></h4>";
        }
    }

    public function tampilan_penjualan_produk() {
        echo "<h2 class='info'>Jumlah Penjualan Tahun Ini</h2>";
    }
}
```

```

$tahun = date("Y");

$jumlahLele = mysql_query("SELECT SUM(IF(tahun = '$tahun',
    jumlah,0)) AS Jumlah_Pertahun FROM _penjualan_lele");
$dataLele = mysql_fetch_array($jumlahLele);
$jmlLele = $dataLele['Jumlah_Pertahun'];

$jumlahNilai = mysql_query("SELECT SUM(IF(tahun = '$tahun',
    jumlah,0)) AS Jumlah_Pertahun FROM _penjualan_nilai");
$dataNilai = mysql_fetch_array($jumlahNilai);
$jmlNilai = $dataNilai['Jumlah_Pertahun'];

$jumlahWader = mysql_query("SELECT SUM(IF(tahun = '$tahun',
    jumlah,0)) AS Jumlah_Pertahun FROM _penjualan_wader");
$dataWader = mysql_fetch_array($jumlahWader);
$jmlWader = $dataWader['Jumlah_Pertahun'];

echo "<h4 class='isi_informasi'>
    <div id='spasi'>Penjualan Keripik Lele : $jmlLele</div>
    <div id='spasi'>Penjualan Keripik Nilai : $jmlNilai</div>
    <div id='spasi'>Penjualan Keripik Wader: $jmlWader</div>
</h4>";
}

public function tampilan_ulang_tahun_bulan_ini() {
    echo "<h2 class='info'>Daftar Ulang Tahun Pelanggan
        Bulan Ini</h2>";

    $bulan = date("M");

    $stampil = mysql_query("SELECT nama_pelanggan, tgl_lahir
        FROM _pelanggan WHERE tgl_lahir LIKE '%$bulan%' ORDER BY
        tgl_lahir DESC");
    while ($a=mysql_fetch_array($stampil)) {
        echo "<h4 class='isi_informasi'><div id='spasi'>
            $a[nama_pelanggan]</div><div id='spasi_ultra'>
            $a[tgl_lahir]</div></h4>";
    }
}

public function tampilan_grafik_penjualan_pertahun() {
    echo "<h2 class='info'>Grafik Penjualan
        Produk Pertahun</h2>";
    echo "<h4 class='isi_informasi'>";
    include "grafikPenjualan.php"; echo "</h4>";
}

public function crud_beranda() {
    $get_data = new modul_beranda();

    switch ($_GET[data]) {
        default:
            $get_data->tampilan_selamat_datang();
            $get_data->tampilan_informasi_baru();
            $get_data->tampilan_ulang_tahun_bulan_ini();
    }
}

```

```

        $get_data->tampilan_grafik_penjualan_pertahun();
        $get_data->tampilan_penjualan_produk();
        break;
    }
}
?>

```

diagram_penjualan.php

```

<?php
class pilihan_dari_combobox {
    public function pilih_keripik() {
        echo "<script language='Javascript'>
        function pilih(id) {
            if (id=='PR01') {
                location.replace('?module=diagram&data=
                tampilan_diagram_penjualan_lele');
            } else if (id=='PR02') {
                location.replace('?module=diagram&data=
                tampilan_diagram_penjualan_nila');
            } else if (id=='PR03') {
                location.replace('?module=diagram&data=
                tampilan_diagram_penjualan_wader');
            }
        }

        </script>";
    }

    public function pilih_tahun(){

        echo "<script language='Javascript'>
        function tahun(th) {
            location.replace('?module=diagram&data='+th);
        }
        </script>";
    }
}

class modul_diagram_penjualan extends pilihan_dari_combobox {
    public function tampilan_diagram_penjualan_lele() {
        $sambil_fungsi = new pilihan_dari_combobox();
        $sambil_fungsi->pilih_keripik();

        echo "<section id='konten'><h2 class='kotak'>
        Diagram Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan Keripik
        Lele Perbulan</h2>
        <h4 class='isi_informasi'>";
        include "penjualan_lele.php"; echo "</h4>";
        echo "<table><tr><td class='combobox'>
        <select name='produk' onChange='pilih(this.value)'
        class='field size2'><option value=0 selected>
        - Pilih Produk -</option>";
    }
}

```

```

$Produk = mysql_query("SELECT * FROM _produk ORDER BY
    id_produk");
while ($a=mysql_fetch_array($Produk)) {
    echo "<option value='$a[id_produk] '$a[produk]</option>";
}

echo "</select></td></tr></table></section>";
}

public function tampilan_diagram_penjualan_nila() {
    $ambil_fungsi = new pilihan_dari_combobox();
    $ambil_fungsi->pilih_keripik();

    echo "<section id='konten'><h2 class='kotak'>
        Diagram Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan Keripik
        Nila Perbulan</h2>
        <h4 class='isi_informasi'>";
    include "penjualan_nila.php"; echo "</h4>";
    echo "<table><tr><td class='combobox'>
        <select name='produk' onChange='pilih(this.value)'
        class='field size2'><option value=0 selected>
        - Pilih Produk -</option>";
    $Produk = mysql_query("SELECT * FROM _produk ORDER BY
        id_produk");
    while ($a=mysql_fetch_array($Produk)) {
        echo "<option value='$a[id_produk] '$a[produk]</option>";
    }

    echo "</select></td></tr></table></section>";
}

public function tampilan_diagram_penjualan_wader() {
    $ambil_fungsi = new pilihan_dari_combobox();
    $ambil_fungsi->pilih_keripik();

    echo "<section id='konten'><h2 class='kotak'>
        Diagram Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan Keripik
        Wader Perbulan</h2>
        <h4 class='isi_informasi'>";
    include "penjualan_wader.php"; echo "</h4>";
    echo "<table><tr><td class='combobox'><select name='produk'
        onChange='pilih(this.value)' class='field size2'>
        <option value=0 selected>- Pilih Produk -</option>";
    $Produk = mysql_query("SELECT * FROM _produk ORDER BY
        id_produk");
    while ($a=mysql_fetch_array($Produk)) {
        echo "<option value='$a[id_produk] '$a[produk]</option>";
    }

    echo "</select></td></tr></table></section>";
}

public function crud_diagram_penjualan() {
    $get_data = new modul_diagram_penjualan();

```

```

switch ($_GET[data]) {
    default:
        $get_data->tampilan_diagram_penjualan_lele();
        break;
    case 'tampilan_diagram_penjualan_nila':
        $get_data->tampilan_diagram_penjualan_nila();
        break;
    case 'tampilan_diagram_penjualan_wader':
        $get_data->tampilan_diagram_penjualan_wader();
        break;
}
}
}
?>

```

statistik_penjualan.php

```

<?php
class pilihan_dari_combobox {
    public function pilih_keripik(){
        echo "<script language='Javascript1.2'>
        function pilih(id) {
            if (id=='PR01') {
                location.replace('?module=prosentase&data=
                tampilan_statistik_penjualan_lele');
            } else if (id=='PR02') {
                location.replace('?module=prosentase&data=
                tampilan_statistik_penjualan_nila');
            } else if (id=='PR03') {
                location.replace('?module=prosentase&data=
                tampilan_statistik_penjualan_wader');
            }
        }
        </script>";
    }

    public function pilih_tahun() {
        echo "<script language='Javascript1.2'>
        function tahun(th) {
            if (th=='2013') {
                location.replace(<?php mysql_query('SELECT * FROM
                _penjualan_lele WHERE tahun =2013'); ?>)
            }
        }
        </script>";
    }
}

class modul_statistik_penjualan extends pilihan_dari_combobox {
    public function tampilan_statistik_penjualan_lele() {
        $ambil_fungsi = new pilihan_dari_combobox();
        $ambil_fungsi->pilih_keripik();

        echo "<section id='konten'><h2 class='kotak'>

```

```

        Statistik Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan
            Keripik Lele Perbulan</h2>
        <h4 class='isi_informasi'>";
        include "statistik_lele.php"; echo "</h4>";
        echo "<table><tr><td class='combobox'><select name='produk'
            onChange='pilih (this.value)' class='field size2'>
            <option value=0 selected>- Pilih Produk -</option>";
        $Produk = mysql_query("SELECT * FROM _produk ORDER BY
            id_produk");
        while ($a=mysql_fetch_array($Produk)) {
            echo "<option value='$a[id_produk]'>$a[produk]</option>";
        }
        echo "</select></td></tr></table></section>";
    }

public function tampilan_statistik_penjualan_nila(){
    $ambil_fungsi = new pilihan_dari_combobox();
    $ambil_fungsi->pilih_keripik();

    echo "<section id='konten'><h2 class='kotak'>Statistik
        Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan Keripik
            Nila Perbulan</h2>
        <h4 class='isi_informasi'>";
    include "statistik_nila.php";
    echo "</h4>";
    echo "<table>
        <tr><td class='combobox'><select name='produk'
            onChange='pilih(this.value)' class='field size2'>
            <option value=0 selected>- Pilih Produk -</option>";
    $Produk = mysql_query("SELECT * FROM _produk ORDER BY
        id_produk");
    while ($a=mysql_fetch_array($Produk)){
        echo "<option value='$a[id_produk]'>$a[produk]</option>";
    }
    echo "</select></td></tr></table></section>";
}

public function tampilan_statistik_penjualan_wader(){
    $ambil_fungsi = new pilihan_dari_combobox();
    $ambil_fungsi->pilih_keripik();

    echo "<section id='konten'><h2 class='kotak'>Statistik
        Penjualan Produk Perbulan</h2>
        <h2 class='info'>Diagram Jumlah Penjualan Keripik
            Wader Perbulan</h2>
        <h4 class='isi_informasi'>";
    include "statistik_wader.php"; echo "</h4>";
    echo "<table>
        <tr><td class='combobox'><select name='produk'
            onChange='pilih(this.value)' class='field size2'>
            <option value=0 selected>- Pilih Produk -</option>";
    $Produk = mysql_query("SELECT * FROM _produk ORDER BY
        id_produk");

```

```

        while ($a=mysql_fetch_array($Produk)) {
            echo "<option value='$a[id_produk] '$a[produk]</option>";
        }
        echo "</select></td></tr></table></section>";
    }

    public function crud_statistik_penjualan() {
        $get_data = new modul_statistik_penjualan();

        switch ($_GET[data]) {
            default:
                $get_data->tampilan_statistik_penjualan_lele();
                break;
            case 'tampilan_statistik_penjualan_nila':
                $get_data->tampilan_statistik_penjualan_nila();
                break;
            case 'tampilan_statistik_penjualan_wader':
                $get_data->tampilan_statistik_penjualan_wader();
                break;
        }
    }
}
?>

```

set_stok.php

```

<?php

include "_set_modul.php";

class modul_stok_retail extends eksekusi_kontrol{
    public function cari_data_retail() {
        $cari = $_POST['cari'];
        $query = mysql_query("SELECT * FROM _stok_retail WHERE
            nama_retail LIKE '%$cari%'");
        $cek = mysql_num_rows($query);

        if (empty($cari)) {
            echo "<section id='konten'><h2 class ='kotak'>
                Masukkan Nama Retail yang dicari</h2></section>";
        } elseif ($cek > 0) {
            echo "<section id='konten'>
                <h2 class='kotak'>Daftar Stok Retail</h2>
                <form method='POST' action='?module=stok_retail&data=
                    cari_data_retail'>
                <div id='tabel'>
                <table>
                <tr><td class='judul' colspan='7'>
                    Data Stok Retail</td></tr>
                <tr><th>No</th><th>Nama Retail</th><th>
                    Alamat Retail</th><th>Pemilik</th>
                <th>Stok Lele</th><th>Stok Nila</th><th>
                    Stok Wader</th></tr>";

            $nomer=1;
            while ($data=mysql_fetch_array($query)) {

```

```

        echo "<tr>
        <td>$nomer</td>
        <td>$data[nama_retail]</td>
        <td>$data[alamat_rumah]</td>
        <td>$data[pemilik]</td>
        <td>$data[stok_lele]</td>
        <td>$data[stok_nila]</td>
        <td>$data[stok_wader]</td>
        </tr>";
        $nomer++;
    }
    echo "</table></div></section>";
} else {
    echo "<section id='konten'><h2 class='kotak'>
        Tidak ada data dengan kata $cari</h2></section>";
}
}

public function tampil_data_stok_retail() {
    echo "<section id='konten'>
    <h2 class='kotak'>Daftar Stok Retail</h2><form method='POST'
        action='?module=stok_retail&data=cari_data_retail'>
    <input type='text' class='field' name='cari' placeholder=
        'Cari Data Stok Retail' >
    <input type='submit' class='button' value='Cari' /></form>
    <div id='tabel'>
    <table>
    <tr><td class='judul' colspan='8'>Data Stok Retail</td></tr>
    <tr><th>No</th><th>Nama Retail</th><th>Alamat
        Retail</th><th>Pemilik</th>
    <th>Stok Lele</th><th>Stok Nila</th><th>StokWader</th>
    <th>Aksi</th></tr>";

    $page = new halaman();
    $batas = 10;
    $posisi = $page->cariPosisi($batas);

    $nomer=$posisi+1;
    $stampil = mysql_query("SELECT * FROM _stok_retail ORDER BY
        id_stok LIMIT $posisi,$batas");
    while ($data=mysql_fetch_array($stampil)){
        echo "<tr>
        <td>$nomer</td>
        <td width='150'>$data[nama_retail]</td>
        <td width='200'>$data[alamat_retail]</td>
        <td width='200'>$data[pemilik]</td>

        <td width='50'>$data[stok_lele]</td>
        <td width='50'>$data[stok_nila]</td>
        <td width='50'>$data[stok_wader]</td>
        <td align='center'><a href=?module=stok_retail
            &data=edit_data_stok_retail&id=$data[id_stok]>
            <img src=../../Desain/edit.png></a>
        </td>";
        $nomer++;
    }
}

```

```

echo "</table>";
$view = mysql_num_rows(mysql_query("SELECT * FROM
    _stok_retail"));

$jmlHalaman = $page->jumlahHalaman($view, $batas);
$linkHalaman = $page->navHalaman($_GET['halaman'],
    $jmlHalaman);
echo "$linkHalaman";
echo "</div></section>";
}

public function edit_data_stok_retail() {
    $edit = mysql_query("SELECT * FROM _stok_retail WHERE
        id_stok = '$_GET[id]'");
    $urut = mysql_fetch_array($edit);

    echo "<section id='konten'>
        <h2 class='kotak'>Edit Data Stok Retail</h2>
        <h2 class='info'>Edit Data Stok Retail</h2>
        <div class=form> <form method='POST' action='../aplikasi/
            DaftarStokRetail/_set_control.php?module=
            stok_retail&data=updatestokretail'>
        <input type='hidden' name='id' value='$_urut[id_stok]'>
        <div id='dalamForm'>
        <table>
        <tr><td>Nama Retail</td><td> :
        <select name='nama_retail' disabled class='field size2'
            value='$_urut[nama_retail]'><option value=0 selected>-
            Pilih Retail -</option>";

    $retail = mysql_query("SELECT * FROM _pelanggan ORDER BY
        id_pelanggan");
    while ($a=mysql_fetch_array($retail)) {
        if ($urut[nama_retail]==$a[nama_retail]) {
            echo "<option value='$a[nama_retail]' selected>$a
                [nama_retail]</option>";
        } else {
            echo "<option value='$a[nama_retail]'>$a
                [nama_retail]</option>";
        }
    }
}
echo "</td></tr></select>
    <tr><td>Alamat Retail</td><td>&nbsp;&nbsp;&nbsp;<textarea
        name='alamat_retail' disabled class='field size3'>
        $_urut[alamat_retail]</textarea></td></tr>
    <tr><td>Pemilik</td><td>: <input type='text' disabled
        name='pemilik' class='field size1' value=
        '$_urut[pemilik]'></td></tr>
    <tr><td>Stok Lele</td><td>:<input type='text' name=
        'stok_lele' class='field size1' value=
        '$_urut[stok_lele]'></td></tr>
    <tr><td>Stok Nila</td><td> : <input type='text' name=
        'stok_nila' class='field size1' value=
        '$_urut[stok_nila]'></td></tr>
    <tr><td>Stok Wader</td><td> : <input type='text'
        name='stok_wader' class='field size1'

```

```

        value='$urut[stok_wader]}'></td></tr>
<tr><td colspan=2><input type=submit value class='update'>
<input type=button class='batal'onclick=self.history.back()>
</td></tr></table></div></form></div></section>";
}

public function update_stok_retail() {
mysql_query("UPDATE _stok_retail SET stok_lele = '$_POST
[stok_lele]', stok_nila= '$_POST[stok_nila]',stok_wader =
'$_POST[stok_wader]' WHERE id_stok= '$_POST[id]'");
}

public function crud_stok_retail(){
$get_data = new modul_stok_retail();
switch ($_GET[data]) {
    default:
        $get_data->tampil_data_stok_retail();
        break;

    case 'cari_data_retail':
        $get_data->cari_data_retail();
        break;

    case 'tambah_data_stok_retail':
        $get_data->tambah_data_stok_retail();
        break;

    case 'edit_data_stok_retail':
        $get_data->edit_data_stok_retail();
        break;
}
}
}
?>

```

set_control_stok.php

```

<?php
session_start();

include "../konfigurasi/set_init.php";
include '_set_stok.php';

$module = $_GET[module];
$action = $_GET[data];

$get_data = new modul_stok_retail();
$get_data->set_module($module);
$get_data->set_action($action);

if ($get_data->get_module()=='stok_retail' AND $get_data->
get_action()=='inputstokretail') {
    $get_data->simpan_data_stok_retail();
    header('location:../control/administrasi.php?module=

```

```

'.'.$module);
}
elseif ($get_data->get_module()=='stok_retail' AND $get_data->
get_action()=='updatestokretail'){
    $get_data->update_stok_retail();
    header('location:../../control/administrasi.php?module=
'.'.$module);
}
?>

```

set_pesanan.php

```

<?php
include '_set_modul.php';

class modul_pesan extends eksekusi_kontrol
{
    public function cari_data_pesan ()
    {
        $cari      = $_POST['cari'];
        $query      = mysql_query("SELECT * FROM _pesanan WHERE
nama_pelanggan LIKE '%$cari%'");
        $cek      = mysql_num_rows($query);
        $nomer      = 1;

        if (empty($cari))
        {
            echo "<section id='konten'><h2
class='kotak'>Masukkan Nama Pelanggan untuk
mencari</h2></section>";
        }
        elseif ($cek > 0)
        {
            echo "<section id='konten'>
            <h2 class='kotak'>Daftar Pesanan</h2>
            <div id='tabel'>
            <table>

<tr><th></th><th>Tanggal</th><th>Retail</th><th>ID
Transaksi</th><th>Produk</th><th>Qty</th><th>Status</th></tr>";
            while ($data=mysql_fetch_array($query))
            {
                echo "<tr>
                    <td>$nomer</td>
                    <td>$data[tanggal]</td>
                    <td>$data[nama_pelanggan]</td>
                    <td>$data[id_transaksi]</td>
                    <td>$data[produk]</td>
                    <td>$data[qty]</td>
                    <td>$data[status]</td>
                    </tr>";
                $nomer++;
            }
            echo "</table></div></section>";
        }
    }
}

```

```

    }
    else
    {
        echo "<section id='konten'><h2
class='kotak'>Tidak Ditemukan Data dengan kata
$cari</h2></section>";
    }
}

public function tampilan_data_pesanan ()
{
    echo "<section id='konten'>
    <h2 class='kotak'>Daftar Pesanan</h2>
    <form method='POST'
action='?module=pesan&data=cari_data_pesanan'>
        <input type='text' class='field small-field'
name='cari' placeholder='Cari Data Pesanan'>
        <input type='submit' class='button' value='Cari'
/></form>
        <div id='tabel'>
        <table>
        <tr>

            <th>No</th><th>Tanggal</th><th>Jam</th><th>Retail</th><th>ID
Transaksi</th>

            <th>Produk</th><th>Qty</th><th>Harga</th><th>Grand
Total</th><th>Status</th><th>Aksi</th>
            </tr>";

    $page = new halaman();
    $batas = 10;
    $posisi = $page->cariPosisi($batas);

    $nomer=$posisi+1;
    $query = mysql_query("SELECT * FROM _pesanan ORDER BY
id_transaksi DESC LIMIT $posisi, $batas");
    while ($data=mysql_fetch_array($query))
    {
        echo "<tr>
                <td>$nomer</td>
                <td>$data[tanggal]</td>
                <td>$data[jam]</td>
                <td>$data[nama_pelanggan]</td>
                <td>$data[id_transaksi]</td>
                <td>$data[produk]</td>
                <td>$data[qty]</td>
                <td>Rp. $data[satuan_harga]</td>
                <td>$data[total_bayar]</td>
                <td>$data[status]</td>
                <td><a
href=?module=pesan&data=edit_data_pesanan&id=$data[id_transaksi]><img
src=../../Desain/edit.png width=10></a>&nbsp;
                <a
href=?module=pesan&data=detail_data_pesanan&id=$data[id_transaksi]><i
mg src=../../Desain/print.png width=10></a>&nbsp;

```

```

                                <a
href=../aplikasi/DaftarPesan/_set_control.php?module=pesan&data=hap
uspesan&id=$data[id_transaksi]>
                                <img
src=../../Desain/delete.png width=10></a>&nbsp;nbsp;
                                </tr>";
                                $nomer++;
                                }
                                echo "</table>";

                                $view = mysql_num_rows(mysql_query("SELECT * FROM
_pesanan"));

                                $jmlHalaman = $page->jumlahHalaman($view, $batas);
                                $linkHalaman = $page->navHalaman($_GET['halaman'],
$jmlHalaman);

                                echo "$linkHalaman";
                                echo "</div></section>";
                                }

                                public function edit_data_pesan ()
                                {
                                        $edit = mysql_query("SELECT * FROM _pesanan WHERE
id_transaksi = '$_GET[id]'");
                                        $urut = mysql_fetch_array($edit);

                                        echo "<section id='konten'>
                                                <h2 class='kotak'>Edit Data Pesanan</h2>
                                                <h2 class='info'>Edit Status Pesanan</h2>
                                                <form method='POST'
action='../aplikasi/DaftarPesan/_set_control.php?module=pesan&data=
updatepesan'>
                                                        <input type='hidden' name='id'
value='$_urut[id_transaksi]'">
                                                        <div id='dalamForm'>
                                                        <table>
                                                                <tr><td>Nama Karyawan</td><td> <input
type='text' disabled name='nama_pelanggan' class='field size1'
value='$_urut[nama_pelanggan]'"></td></tr>
                                                                <tr><td>ID Transaksi</td><td> <input
type='text' disabled name='id_transaksi' class='field size1'
value='$_urut[id_transaksi]'"></td></tr>
                                                                <tr><td>Produk</td><td> <input
type='text' disabled name='produk' class='field size1'
value='$_urut[produk]'"></td></tr>
                                                                <tr><td>Grand Total</td><td> <input
type='text' disabled name='grand_total' class='field size1'
value='$_urut[total_bayar]'"></td></tr>
                                                                <tr><td>Status</td><td> ";
                                                                if($_urut['status']=='Proses') {
                                                                        echo "<input type='checkbox' name='status' value=
'$_urut[status]' checked> Proses";
                                                                } else {
                                                                        echo "<input type='checkbox' name='status' value=

```



```

public function update_data_pesanan()
{
    mysql_query("UPDATE _pesanan SET status =
'$_POST[status]' WHERE id_transaksi = '$_POST[id]'");

    $status = $_POST['status'];
    $query_pesanan = mysql_query("SELECT * FROM _pesanan
ORDER BY id_transaksi DESC");
    $data_pesanan = mysql_fetch_array($query_pesanan);
    $retail = $data_pesanan['nama_pelanggan'];
    $quantity = $data_pesanan['qty'];
    $produk = $data_pesanan['produk'];

    $query_stok = mysql_query("SELECT * FROM _stok_retail
WHERE nama_retail='$retail'");
    $data_stok = mysql_fetch_array($query_stok);
    $stok_lele = $data_stok['stok_lele'];
    $stok_nila = $data_stok['stok_nila'];
    $stok_wader = $data_stok['stok_wader'];

    $kurang_lele = $stok_lele - $quantity;
    $kurang_nila = $stok_nila - $quantity;
    $kurang_wader = $stok_wader - $quantity;

    echo $stok_lele." ".$stok_nila." ".$stok_wader;
    ?>
<script language=JavaScript>

        var sekarang = new Date();
        var tanggal_sekarang = ('0' +
sekarang.getDate()).slice(-2);
        var bulan =
sekarang.getMonth()+1;
        var bulan_sekarang = ('0' +
bulan).slice(-2);
        var tahun_sekarang =
sekarang.getFullYear();
        var jam = ('0' +
sekarang.getHours()).slice(-2);
        var menit = ('0' +
sekarang.getMinutes()).slice(-2);
        var detik = ('0' +
sekarang.getSeconds()).slice(-2);

        var update = new Date();

        update.setDate(update.getDate()+1);
        var tanggal_besok = ('0' +
update.getDate()).slice(-2);
        var bulan_b =
update.getMonth()+1;
        var bulan_besok = ('0' +bulan_b).slice(-
2);

```

```

                                var tahun_besok          =
update.getFullYear();

                                var waktu_pemulihan =
tahun_besok+bulan_besok+tanggal_besok+jam+menit;
                                <?php if(!isset($_GET['pemulihan'])) { ?>
                                    window.location =
"?module=pesan&data=updatepesan&status=<?php echo $status;
?>&id=<?php echo $retail; ?>&pemulihan="+waktu_pemulihan;
                                    <?php } ?>
                                </script>
<?php
    if(isset($_GET['status'])) {
        $status=$_GET['status'];
    }

    if(isset($_GET['pemulihan'])) {
        if ($status=='Ter kirim' AND $produk=='Keripik Lele') {
            $query=mysql_query("UPDATE _stok_retail SET stok_lele =
            '$kurang_lele', expired_lele = '$_GET[pemulihan]' WHERE
            nama_retail = '$retail'");
            //header('location:../../control/administrasi.php?module='.$_GET['module']);
        ?>
            <script type="text/javascript">
//window.location="../../control/administrasi.php?module=<?php echo
$_GET['module']; ?>&cek=1";
</script>

        }
        elseif ($status=='Ter kirim' AND $produk=='Keripik Nila') {
            mysql_query("UPDATE _stok_retail SET stok_nila =
            '$kurang_nila', expired_nila = '$_GET[pemulihan]' WHERE
            nama_retail ='$retail'");
        }
        header('location:../../control/administrasi.php?module='.$_GET['mod
        ule']);
    }
    elseif ($status=='Ter kirim' AND $produk=='Keripik Wader') {
        mysql_query("UPDATE _stok_retail SET stok_wader =
        '$kurang_wader', expired_wader = '$_GET[pemulihan]' WHERE
        nama_retail ='$retail'");
        header('location:../../control/administrasi.php?module='.$_GE
        T['module']);
    }
}

$bulan_sekarang = date("F");
$tahun_sekarang = date("Y");

$query_lele = mysql_query("SELECT * FROM _penjualan_lele WHERE
    bulan = '$bulan_sekarang' AND tahun = '$tahun_sekarang'");
$update_lele = mysql_num_rows($query_lele);
$data_lele = mysql_fetch_array($query_lele);
$bulanLele = $data_lele['bulan'];

```

```

$tahunLele = $data_lele['tahun'];
$jumlahLele = $data_lele['jumlah'];

$query_nila = mysql_query("SELECT * FROM _penjualan_nila WHERE
    bulan = '$bulan_sekarang' AND tahun = '$tahun_sekarang'");
$update_nila = mysql_num_rows($query_nila);
$data_nila = mysql_fetch_array($query_nila);
$bulanNila = $data_nila['bulan'];
$tahunNila = $data_nila['tahun'];
$jumlahNila = $data_nila['jumlah'];

$query_wader = mysql_query("SELECT * FROM _penjualan_wader WHERE
    bulan = '$bulan_sekarang' AND tahun = '$tahun_sekarang'");
$update_wader = mysql_num_rows($query_wader);
$data_wader = mysql_fetch_array($query_wader);
$bulanWader = $data_wader['bulan'];
$tahunWader = $data_wader['tahun'];
$jumlahWader = $data_wader['jumlah'];

$tambah_lele = $jumlahLele + $quantity;
$tambah_nila = $jumlahNila + $quantity;
$tambah_wader = $jumlahWader + $quantity;

if(isset($_GET['pemulihan'])) {
    if ($status=='Ter kirim' AND $produk=='Keripik Lele') {
        if ($update_lele<=0) {
            mysql_query("INSERT INTO _penjualan_lele (bulan, tahun,
                jumlah) VALUES ('$bulan_sekarang', '$tahun_sekarang', '$quantity')");
        } else {
            mysql_query("UPDATE _penjualan_lele SET jumlah =
                '$tambah_lele' WHERE bulan = '$bulan_sekarang' AND tahun =
                '$tahun_sekarang'");
        }
    }
    elseif ($status=='Ter kirim' AND $produk=='Keripik Nila') {
        if ($update_nila<=0) {
            mysql_query("INSERT INTO _penjualan_nila (bulan, tahun,
                jumlah) VALUES ('$bulan_sekarang', '$tahun_sekarang', '$quantity')");
        } else {
            mysql_query("UPDATE _penjualan_nila SET jumlah = '$tambah_nila'
                WHERE bulan = '$bulan_sekarang' AND tahun = '$tahun_sekarang'");
        }
    }
    elseif ($status=='Ter kirim' AND $produk=='Keripik Wader') {
        if ($update_wader<=0) {
            mysql_query("INSERT INTO _penjualan_wader (bulan, tahun,
                jumlah) VALUES ('$bulan_sekarang', '$tahun_sekarang', '$quantity')");
        } else {
            mysql_query("UPDATE _penjualan_wader SET jumlah =
                '$tambah_wader' WHERE bulan = '$bulan_sekarang' AND tahun =
                '$tahun_sekarang'");
        }
    }
}
}

```

```
public function hapus_data_pesan ()
{
    $sid = $_GET['id'];
    mysql_query("DELETE FROM _pesanan WHERE id_transaksi =
'$sid'");
}

public function crud_pesan ()
{
    $get_data = new modul_pesan();

    switch ($_GET[data])
    {
        default:
            $get_data->tampilan_data_pesan();
            break;

        case 'cari_data_pesan':
            $get_data->cari_data_pesan();
            break;

        case 'edit_data_pesan':
            $get_data->edit_data_pesan();
            break;

        case 'detail_data_pesan':
            $get_data->detail_data_pesan();
            break;
    }
}
}
?>
```

Source Code

Aplikasi Android

login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:scrollbars="vertical">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/background">
        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:background="#16BD00"
            android:gravity="center">
            <ImageView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@drawable/banner_login"
            />
        </LinearLayout>
        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:paddingTop="15dip"
            android:gravity="center">
            <ImageView
                android:layout_width="wrap_content"
                android:layout_gravity="center"
                android:paddingTop="13dip"
                android:layout_height="wrap_content"
                android:src="@drawable/Logo_login"
            />
        </LinearLayout>
        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:paddingTop="35dip"
            android:gravity="center">
            <EditText
                android:layout_width="230dip"
                android:layout_height="wrap_content"
                android:id="@+id/inputemail"
                android:hint="@string/inputemail"
                android:inputType="textEmailAddress"/>
            <EditText
                android:layout_width="230dip"
            />
        </LinearLayout>
    </LinearLayout>
</ScrollView>
```

```

        android:layout_height="wrap_content"
        android:id="@+id/inputpass"
        android:hint="@string/inputpass"
        android:inputType="textPassword"/>

</LinearLayout>
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingTop="35dip"
    android:gravity="center">
    <Button
        android:id="@+id/Login"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:text="@string/masuk"
        android:textStyle="bold"/>
    <Button
        android:id="@+id/daftar"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:text="@string/daftar"
        android:textStyle="bold"/>
</LinearLayout>
</LinearLayout>
</ScrollView>

```

daftar.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:scrollbars="vertical">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="@drawable/background">
        <LinearLayout
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:background="#03C90C"
            android:gravity="center">
            <ImageView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@drawable/banner_daftar"

android:contentDescription="@drawable/banner_daftar"/>
            </LinearLayout>

```

```

<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingTop="15dip"
    android:gravity="center">
    <ImageView
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:paddingTop="13dip"
        android:layout_height="wrap_content"
        android:src="@drawable/logo_daftar"
        android:contentDescription="@drawable/Logo_daftar"/>
    </LinearLayout>
    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:paddingTop="35dip"
        android:gravity="center">
        <EditText
            android:layout_width="230dip"
            android:layout_height="wrap_content"
            android:id="@+id/reg_nama"
            android:hint="@string/reg_nama"/>
        <EditText
            android:layout_width="230dip"
            android:layout_height="wrap_content"
            android:id="@+id/reg_email"
            android:hint="@string/reg_email"
            android:inputType="textEmailAddress"/>
        <EditText
            android:layout_width="230dip"
            android:layout_height="wrap_content"
            android:id="@+id/reg_pass"
            android:hint="@string/reg_password"
            android:inputType="textPassword"/>
        <Button
            android:id="@+id/reg_Lahir"
            android:layout_width="230dip"
            android:layout_height="wrap_content"
            android:text="@string/reg_Lahir"
            android:textStyle="bold"/>
        <TextView
            android:id="@+id/dateAndTimeLabel"
            android:layout_width="230dip"
            android:layout_height="wrap_content"
            android:textStyle="bold"
            android:textColor="#8C8B99"
            android:gravity="center"/>

        <EditText
            android:layout_width="230dip"

```

```

        android:layout_height="wrap_content"
        android:layout_margin="5dip"
        android:layout_marginBottom="15dip"
        android:lines="4"
        android:gravity="top"
        android:id="@+id/reg_alamat"
        android:hint="@string/reg_alamat"/>
<Spinner
    android:id="@+id/reg_jenis_kelamin"
    android:layout_width="230dip"
    android:layout_height="wrap_content"/>

<Spinner
    android:id="@+id/reg_agama"
    android:layout_width="230dip"
    android:layout_height="wrap_content"/>
<EditText
    android:layout_width="230dip"
    android:layout_height="wrap_content"
    android:id="@+id/reg_retail"
    android:hint="@string/reg_retail"/>
<EditText
    android:layout_width="230dip"
    android:layout_height="wrap_content"
    android:layout_margin="5dip"
    android:layout_marginBottom="15dip"
    android:lines="4"
    android:gravity="top"
    android:id="@+id/reg_retail_alamat"
    android:hint="@string/reg_retail_alamat"/>
<EditText
    android:layout_width="230dip"
    android:layout_height="wrap_content"
    android:id="@+id/reg_telp"
    android:hint="@string/reg_telp"
    android:inputType="text|phone"/>

</LinearLayout>
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingTop="35dip"
    android:gravity="center">
    <Button
        android:id="@+id/simpan"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:text="@string/simpan"
        android:textStyle="bold" />
    <Button
        android:id="@+id/batal"
        android:layout_width="200dip"

```

```

                android:layout_height="wrap_content"
                android:text="@string/batal"
                android:textStyle="bold" />
            </LinearLayout>
        </LinearLayout>
    </ScrollView>

```

tab_setting.xml

```

<?xml version="1.0" encoding="utf-8"?>
<TabHost
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@android:id/tabhost"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent">
        <TabWidget
            android:id="@android:id/tabs"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"/>
        <FrameLayout
            android:id="@android:id/tabcontent"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"/>
    </LinearLayout>
</TabHost>

```

main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center|center_horizontal"
    android:background="@drawable/splash_in">
    <ProgressBar
        android:id="@+id/progressBar2"
        style="?android:attr/progressBarStyleSmall"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="250dip"/>
</LinearLayout>

```

beranda.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"

```

```

android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:scrollbars="vertical">
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:orientation="horizontal"
  android:padding="4dip">
  <ImageView
    android:id="@+id/icon"
    android:layout_width="wrap_content"
    android:paddingLeft="10dip"
    android:paddingRight="10dip"
    android:paddingTop="13dip"
    android:layout_height="wrap_content"
    android:src="@drawable/view_info"
    android:contentDescription="@drawable/view_info"/>
    <LinearLayout
      android:layout_width="fill_parent"
      android:layout_height="wrap_content"
      android:orientation="vertical">
      <TextView
        android:id="@+id/judul"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:textStyle="bold"
        android:textColor="#1382E1"/>
      <TextView
        android:id="@+id/tanggal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="15sp"
        android:textStyle="bold"
        android:textColor="#1382E1"/>
      <TextView
        android:id="@+id/deskripsi"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
    </LinearLayout>
  </LinearLayout>
</ScrollView>

```

produk_item.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:orientation="horizontal"
  android:padding="4dip">
  <ImageView

```

```

        android:id="@+id/icon"
        android:layout_width="wrap_content"
        android:paddingLeft="10dip"
        android:paddingRight="10dip"
        android:paddingTop="13dip"
        android:layout_height="wrap_content"
        android:src="@drawable/view_produk"
        android:contentDescription="@drawable/view_produk"/>
<RelativeLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <TextView
            android:id="@+id/nama_produk"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textSize="20sp"
            android:textStyle="bold"/>
        <TextView
            android:id="@+id/harga_produk"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textSize="15sp"
            android:textStyle="bold"/>
    </LinearLayout>
    <TextView
        android:id="@+id/id_produk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_centerVertical="true"
        android:textColor="#000000"/>

</RelativeLayout>
</LinearLayout>

```

produk_pesanan.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:background="@drawable/background">
    <LinearLayout
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:background="#03C90C">
        <ImageView

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/banner_pesan"

android:contentDescription="@drawable/banner_pesan"/>
</LinearLayout>
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingTop="30dip">
    <EditText
        android:id="@+id/nama_produk"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:hint=""/>
    <EditText
        android:id="@+id/qty"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:hint="@string/qty"
        android:layout_gravity="center"
        android:inputType="number"/>
    <EditText
        android:id="@+id/retail"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:hint="@string/namaretail"
        android:layout_gravity="center"/>
</LinearLayout>
<LinearLayout
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:paddingTop="30dip">
    <Button
        android:id="@+id/pesan"
        android:layout_width="200dip"
        android:layout_height="wrap_content"
        android:text="@string/pesan"
        android:textStyle="bold"
        android:layout_gravity="center"/>
</LinearLayout>
</LinearLayout>

```

keluar.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center|center_horizontal"

```

```

        android:background="@drawable/splash_out">
        <ProgressBar
            android:id="@+id/progressBar2"
            style="?android:attr/progressBarStyleSmall"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:paddingTop="250dip"/>
    </LinearLayout>

```

manifest.xml

```

<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.babyfishbeta"
    android:versionCode="1"
    android:versionName="1.0" >

    <supports-screens
        android:largeScreens="true"
        android:normalScreens="true"
        android:smallScreens="true"
        android:resizeable="true"
        android:anyDensity="true"/>

    <uses-sdk android:minSdkVersion="8" android:targetSdkVersion="17"/>
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.RECEIVE_SMS"/>
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.GET_ACCOUNTS"/>
    <uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE"/>

    <application android:icon="@drawable/ic_launcher"
android:label="@string/app_name" >
        <activity android:name=".BabyFishActivityBeta"
android:label="@string/app_name" >
            <intent-filter> <action
android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".Beranda"></activity>
        <activity android:name=".Produk"></activity>
        <activity android:name=".Histori"></activity>
        <activity android:name=".Profil"></activity>
        <activity android:name=".Bantuan"></activity>
        <activity android:name=".TabSetting"></activity>
        <activity android:name=".ProdukPesan"></activity>
        <activity android:name=".Petunjuk"></activity>
        <activity android:name=".Tentang"></activity>
    </application>

```

```

        <activity android:name=".InformasiPembayaran"></activity>
        <activity android:name=".Keluar"></activity>
        <activity android:name=".Login"></activity>
        <activity android:name=".Daftar"></activity>
    </application>
</manifest>

```

BabyFishActivity.java

```

package com.babyfishbeta;

import android.app.Activity;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.MotionEvent;

public class BabyFishActivityBeta extends Activity {
    protected boolean _active = true;
    protected int _splashTime = 2000;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        setContentView(R.layout.main);

        // thread untuk menampilkan SplashScreen
        Thread splashTread = new Thread() {
            @Override
            public void run() {
                try {
                    int waited = 0;
                    while(_active && (waited < _splashTime)) {
                        sleep(100);
                        if(_active) {
                            waited += 100;
                        }
                    }
                } catch (InterruptedException e) {
                    // kosong saja
                } finally {
                    finish();
                    Intent newIntent=new
Intent (BabyFishActivityBeta.this,Login.class);
                    startActivityForResult (newIntent,0);
                }
            }
        };
        splashTread.start();
    }

    @Override
    public boolean onTouchEvent (MotionEvent event) {

```

```

        login = (Button) findViewById(R.id.login);
        login.setOnClickListener(this);
        daftar = (Button) findViewById(R.id.daftar);
        daftar.setOnClickListener(this);
    }

    public void _login() {
        try {
            httpClient = new DefaultHttpClient();
            httpPost = new
HttpPost("http://www.babyfish.hol.es/server_bf/Android/cek_pelangga
n.php");
            nameValuePairs = new ArrayList<NameValuePair>(2);
            nameValuePairs.add(new
BasicNameValuePair("email",
usernameEditText.getText().toString().trim()));
            nameValuePairs.add(new
BasicNameValuePair("password",
passwordEditText.getText().toString().trim()));
            httpPost.setEntity(new
UrlEncodedFormEntity(nameValuePairs));
            httpResponse = httpClient.execute(httpPost);
            ResponseHandler<String> responseHandler = new
BasicResponseHandler();
            final String string =
httpClient.execute(httpPost, responseHandler);
            System.out.println("Response : " +string);
            runOnUiThread(new Runnable() {

                public void run() {
                    Toast.makeText(Login.this, "Keterangan
: " +string, Toast.LENGTH_SHORT).show();
                }
            });

            if (string.equalsIgnoreCase("Ok")) {
                runOnUiThread(new Runnable() {

                    public void run() {
                        Toast.makeText(Login.this,
"Masuk", Toast.LENGTH_SHORT).show();
                    }
                });

                startActivity(new Intent(Login.this,
TabSetting.class));
            } else {
                showAlert();
            }
        } catch (Exception e) {
            dialog.dismiss();
            System.out.println("Keterangan : "+
e.getMessage());
        }
    }
}

```

```

public void onClick (View panggil) {
switch (panggil.getId()) {
case R.id.login:
login.setOnClickListener(new OnClickListener() {

public void onClick(View v) {
_login();

}

});
break;

case R.id.daftar:
Intent _daftar = new Intent (panggil.getContext(),
Daftar.class);
startActivityForResult(_daftar, 0);
break;

default:
break;
}
}

public void showAlert() {
Login.this.runOnUiThread(new Runnable() {

public void run() {
AlertDialog.Builder builder = new
AlertDialog.Builder(Login.this);
builder.setTitle("Login Error");
builder.setMessage("Email / Password
Salah");

builder.setPositiveButton("Ok", new
DialogInterface.OnClickListener() {

public void onClick(DialogInterface
arg0, int arg1) {

// TODO Auto-generated method
stub

}

});
builder.show();
}

});
}
}

```

daftar.java

```

package com.babyfishbeta;

import java.util.ArrayList;
import java.util.Calendar;
import java.text.DateFormat;

import org.apache.http.NameValuePair;

```

```

import org.apache.http.message.BasicNameValuePair;

import android.app.Activity;
import android.app.DatePickerDialog;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

public class Daftar extends Activity implements
OnItemSelectedListener {

    EditText namaPelangganEditText;
    EditText emailEditText;
    EditText passwordEditText;
    EditText alamatRumahEditText;
    EditText namaRetailEditText;
    EditText alamatRetailEditText;
    EditText noTelpEditText;
    Button simpanButton;
    Button batalButton;
    Spinner jenisKelaminSpinner;
    Spinner agamaSpinner;
    Button lahirButton;

    private String[] jenisKelaminPilihan = {"- Pilih Jenis
Kelamin -",
Laki - Laki ",
Perempuan " };
    private String[] agamaPilihan = {"- Pilih Agama -",
" Budha ",
" Hindu ",
" Islam",
"
Katholik",
" Kristen",
"
Khonghuchu" };

    DateFormat dateFormat = DateFormat.getDateTimeInstance();
    TextView dateAndTimeLabelTextView;
    Calendar dateAndTime = Calendar.getInstance();
    DatePickerDialog.OnDateSetListener d = new
DatePickerDialog.OnDateSetListener() {

```

```

        public void onDateSet(DatePicker view, int year, int
month, int day) {
            dateAndTime.set(Calendar.YEAR,year);
            dateAndTime.set(Calendar.MONTH,month);
            dateAndTime.set(Calendar.DAY_OF_MONTH,day);
            updateLabel();
        }
    };

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        try {
            setContentView(R.layout.daftar);
            jenisKelaminSpinner =
(Spinner) findViewById(R.id.reg_jenis_kelamin);
            @SuppressWarnings({ "unchecked", "rawtypes" })
            ArrayAdapter jk = new ArrayAdapter(this,
android.R.layout.simple_spinner_item,jenisKelaminPilihan);

            jk.setDropDownViewResource(android.R.layout.simple_spinner_dr
opdown_item);
            jenisKelaminSpinner.setAdapter(jk);

            jenisKelaminSpinner.setOnItemClickListener(this);

            agamaSpinner =
(Spinner) findViewById(R.id.reg_agama);
            @SuppressWarnings({ "unchecked", "rawtypes" })
            ArrayAdapter ag = new ArrayAdapter(this,
android.R.layout.simple_spinner_item,agamaPilihan);

            ag.setDropDownViewResource(android.R.layout.simple_spinner_dr
opdown_item);
            agamaSpinner.setAdapter(ag);
            agamaSpinner.setOnItemClickListener(this);
            namaPelangganEditText =
(EditText) findViewById(R.id.reg_nama);
            emailEditText =
(EditText) findViewById(R.id.reg_email);
            passwordEditText =
(EditText) findViewById(R.id.reg_pass);
            lahirButton =
(Button) findViewById(R.id.reg_lahir);
            lahirButton.setOnClickListener(new
OnClickListener() {

                public void onClick(View v) {
                    switch (v.getId()) {
                        case R.id.reg_lahir:
                            settingTanggal();
                            break;
                        default:
                            break;
                    }
                }
            });
        }
    };

```



```

        res = res.replaceAll("\\s",
");
        if(res.equals("1")) {
            Toast.makeText(Daftar.this, "Gagal Simpan",
Toast.LENGTH_LONG).show();
        }
        else {
            Toast.makeText(Daftar.this, "Data Tersimpan Ke Server",
Toast.LENGTH_LONG).show();
        }
        } catch (Exception e) {
            Toast.makeText(Daftar.this,
"Error : "+ e.toString(), Toast.LENGTH_LONG).show();
        }
    });
}
catch (Exception e) {
    Toast.makeText(Daftar.this, "Error : "
+e.getMessage(), Toast.LENGTH_LONG).show();
}

batalButton = (Button)findViewById(R.id.batal);
batalButton.setOnClickListener(new OnClickListener() {

    public void onClick(View v) {
        Intent intent = new Intent();
        setResult(RESULT_CANCELED, intent);
        finish();
    }

});

public void onItemClickSelectedJk(AdapterView<?> jk, View v, int
posisi, long id) {
    Toast.makeText(Daftar.this,
jenisKelaminPilihan[posisi].toString(), Toast.LENGTH_SHORT).show();
}

public void onNothingSelectedJk(AdapterView<?> jk) {
    Toast.makeText(Daftar.this, "Gak Ada yg Dipilih",
Toast.LENGTH_SHORT).show();
}

public void onItemClickSelectedAg(AdapterView<?> ag, View v, int
posisi, long id) {
    Toast.makeText(Daftar.this,
agamaPilihan[posisi].toString(), Toast.LENGTH_SHORT).show();
}

public void onNothingSelectedAg(AdapterView<?> ag) {
    Toast.makeText(Daftar.this, "Gak Ada yg Dipilih",
Toast.LENGTH_SHORT).show();
}
}

```

```

        public void onItemSelected(AdapterView<?> arg0, View arg1,
int arg2,
                long arg3) {
            // TODO Auto-generated method stub
        }

        public void onNothingSelected(AdapterView<?> arg0) {
            // TODO Auto-generated method stub
        }

        private void settingTanggal() {
            new DatePickerDialog(Daftar.this, d,
                dateAndTime.get(Calendar.YEAR),
                dateAndTime.get(Calendar.MONTH),
                dateAndTime.get(Calendar.DAY_OF_MONTH))
                .show();
        }

        private void updateLabel() {

            dateAndTimeLabelTextView.setText(dateFormat.format(dateAndTime.getTime()));
        }
    }
}

```

TabSetting.java

```

package com.babyfishbeta;

import com.babyfishbeta.R;

import android.app.TabActivity;
import android.os.Bundle;
import android.content.Intent;
import android.widget.TabHost;
import android.widget.TabHost.TabSpec;

public class TabSetting extends TabActivity {

    TabHost tabHost;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.tab_setting);
        tabHost = getTabHost();

        TabSpec beranda = tabHost.newTabSpec("Beranda");
        beranda.setIndicator("Beranda",
getResources().getDrawable(R.drawable.style_beranda));
        Intent berandaIntent = new Intent(this,
Beranda.class);
        beranda.setContent(berandaIntent);

        TabSpec produk = tabHost.newTabSpec("Produk");
    }
}

```

```

        produk.setIndicator("Produk",
getResources().getDrawable(R.drawable.style_produk));
        Intent produkIntent = new Intent(this,
Produk.class);
        produk.setContent(produkIntent);

        TabSpec histori = tabHost.newTabSpec("Histori");
        histori.setIndicator("Histori",
getResources().getDrawable(R.drawable.style_histori));
        Intent historiIntent = new Intent(this,
Histori.class);
        histori.setContent(historiIntent);

        TabSpec bantuan = tabHost.newTabSpec("Bantuan");
        bantuan.setIndicator("Bantuan",
getResources().getDrawable(R.drawable.style_bantuan));
        Intent aboutIntent = new Intent(this,
Bantuan.class);
        bantuan.setContent(aboutIntent);

        tabHost.addTab(beranda);
        tabHost.addTab(produk);
        tabHost.addTab(histori);
        tabHost.addTab(bantuan);
    }
}

```

beranda.java

```

package com.babyfishbeta;

import java.util.ArrayList;
import java.util.HashMap;

import com.babyfishbeta.JSONParser;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.ListActivity;
import android.os.Bundle;
import android.widget.ListAdapter;
import android.widget.SimpleAdapter;

public class Beranda extends ListActivity {
    private static String URL =
"http://www.babyfish.hol.es/server_bf/Android/tampil_informasi.php"
;
    private String TAG_DAFTAR_INFORMASI = "daftar_informasi";
    private String TAG_JUDUL = "judul";
    private String TAG_TANGGAL = "tanggal";
    private String TAG_DESKRIPSI = "deskripsi";

    JSONArray contacts = null;
    JSONParser parser = new JSONParser();

```

```

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.beranda);
        ArrayList<HashMap<String, String>> contactList = new
ArrayList<HashMap<String, String>>();
        JSONObject json = parser.getJSONFromUrl(URL);
        try {
            contacts =
json.getJSONArray(TAG_DAFTAR_INFORMASI);
            for (int i = 0; i < contacts.length(); i++) {
                JSONObject data =
contacts.getJSONObject(i);
                String judulString =
data.getString(TAG_JUDUL);
                String tanggalString =
data.getString(TAG_TANGGAL);
                String deskripsiString =
data.getString(TAG_DESKRIPSI);
                HashMap<String, String> map = new
HashMap<String, String>();
                map.put(TAG_JUDUL, judulString);
                map.put(TAG_TANGGAL, tanggalString);
                map.put(TAG_DESKRIPSI, deskripsiString);
                contactList.add(map);
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }

        // Tampilkan Ke ListAdapter
        ListAdapter adapter = new SimpleAdapter(this,
contactList, R.layout.beranda_konten,
            new String[] { TAG_JUDUL, TAG_TANGGAL,
TAG_DESKRIPSI },
            new int[] { R.id.judul, R.id.tanggal,
R.id.deskripsi });
        setListAdapter(adapter);
    }
}

```

produk.java

```

package com.babyfishbeta;

import java.util.ArrayList;
import java.util.HashMap;

import com.babyfishbeta.JSONParser;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import android.app.ListActivity;
import android.content.Intent;

```

```

import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ListAdapter;
import android.widget.ListView;
import android.widget.SimpleAdapter;
import android.widget.TextView;

public class Produk extends ListActivity {
    private static String URL =
"http://www.babyfish.hol.es/server_bf/Android/tampil_produk.php";
    private String TAG_DAFTAR_PRODUK = "daftar_produk";
    private String TAG_NAMA_PRODUK = "produk";
    private String TAG_HARGA_PRODUK = "harga";
    private String TAG_ID_PRODUK = "id_produk";

    JSONArray contacts = null;
    JSONParser parser = new JSONParser();

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.produk);

        fungsi_produk();
    }

    public void fungsi_produk() {
        ArrayList<HashMap<String, String>> contactList = new
ArrayList<HashMap<String, String>>();
        JSONObject json = parser.getJSONFromUrl(URL);
        try {
            contacts =
json.getJSONArray(TAG_DAFTAR_PRODUK);
            for (int i = 0; i < contacts.length(); i++) {
                JSONObject data =
contacts.getJSONObject(i);
                String namaProdukString =
data.getString(TAG_NAMA_PRODUK);
                String hargaProdukString =
data.getString(TAG_HARGA_PRODUK);
                String idProdukString =
data.getString(TAG_ID_PRODUK);
                HashMap<String, String> map = new
HashMap<String, String>();
                map.put(TAG_NAMA_PRODUK,
namaProdukString);
                map.put(TAG_HARGA_PRODUK, hargaProdukString);
                map.put(TAG_ID_PRODUK, idProdukString);
                contactList.add(map);
            }
        } catch (JSONException e) {
            e.printStackTrace();
        }

        ListAdapter adapter = new SimpleAdapter(this,

```



```

JSONParser parser = new JSONParser();

public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.produk);

    fungsi_produk();
}

public void fungsi_produk() {
    ArrayList<HashMap<String, String>> contactList = new
ArrayList<HashMap<String, String>>();
    JSONObject json = parser.getJSONFromUrl(URL);
    try {
        contacts =
json.getJSONArray(TAG_DAFTAR_PRODUK);
        for (int i = 0; i < contacts.length(); i++) {
            JSONObject data =
contacts.getJSONObject(i);
            String namaProdukString =
data.getString(TAG_NAMA_PRODUK);
            String hargaProdukString =
data.getString(TAG_HARGA_PRODUK);
            String idProdukString =
data.getString(TAG_ID_PRODUK);
            HashMap<String, String> map = new
HashMap<String, String>();
            map.put(TAG_NAMA_PRODUK,
namaProdukString);
            map.put(TAG_HARGA_PRODUK, hargaProdukString);
            map.put(TAG_ID_PRODUK, idProdukString);
            contactList.add(map);
        }
    } catch (JSONException e) {
        e.printStackTrace();
    }

    ListAdapter adapter = new SimpleAdapter(this,
contactList,
        R.layout.produk_item, new String[] {
TAG_NAMA_PRODUK, TAG_HARGA_PRODUK, TAG_ID_PRODUK },
        new int[] { R.id.nama_produk,
R.id.harga_produk, R.id.id_produk });
    setListAdapter(adapter);
    ListView listView = getListView();
    listView.setOnItemClickListener(new
OnItemClickListener() {

        public void onItemClick(AdapterView<?> data,
View v, int posisi, long id) {
            String string =
((TextView)v.findViewById(R.id.id_produk)).getText().toString();
            Intent intent = new Intent(Produk.this,
ProdukPesan.class);
            intent.putExtra(TAG_ID_PRODUK, string);
            startActivity(intent);
        }
    }
}

```

```

    });
}
}

```

keluar.java

```

package com.babyfishbeta;

import android.app.Activity;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.MotionEvent;

public class Keluar extends Activity {
    protected boolean _active = true;
    protected int _splashTime = 3000;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
        setContentView(R.layout.keluar);

        // thread untuk menampilkan SplashScreen
        Thread splashTread = new Thread() {
            @Override
            public void run() {
                try {
                    int waited = 0;
                    while(_active && (waited < _splashTime)) {
                        sleep(100);
                        if(_active) {
                            waited += 100;
                        }
                    }
                } catch (InterruptedException e) {
                    // kosong saja
                } finally {
                    finish();
                    Intent newIntent=new
Intent (Keluar.this,Login.class);
                    startActivityForResult (newIntent,0);
                }
            }
        };
        splashTread.start();
    }

    @Override
    public boolean onTouchEvent(MotionEvent event) {
        if (event.getAction() == MotionEvent.ACTION_DOWN) {
            _active = false;
        }
    }
}

```

```
        return true;
    }
}
```

JSONParser.java

```
package com.babyfishbeta;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONException;
import org.json.JSONObject;
import android.util.Log;

public class JSONParser {
    static InputStream is = null;
    static JSONObject jsonObj = null;
    static String json = "";
    public JSONParser() {
    }

    public JSONObject getJSONFromUrl(String url) {
        try {
            DefaultHttpClient httpClient = new
DefaultHttpClient();
            HttpPost httpPost = new HttpPost(url);
            HttpResponse httpResponse =
httpClient.execute(httpPost);
            HttpEntity httpEntity =
httpResponse.getEntity();
            is = httpEntity.getContent();
        } catch (UnsupportedEncodingException e) {
            // TODO: handle exception
            e.printStackTrace();
        } catch (ClientProtocolException e) {
            // TODO: handle exception
            e.printStackTrace();
        } catch (IOException e) {
            // TODO: handle exception
            e.printStackTrace();
        }

        try {
            BufferedReader reader = new BufferedReader(new
InputStreamReader(is, "iso-8859-1"), 8);
            StringBuilder sb = new StringBuilder();
            String line = null;
            while ((line = reader.readLine()) != null) {
```

```
        sb.append(line + "\n");
    }
    is.close();
    json = sb.toString();
} catch (Exception e) {
    // TODO: handle exception
    Log.e("Buffer Error", "Error converting result"
+ e.toString());
}

    try {
        jsonObj = new JSONObject(json);
    } catch (JSONException e) {
        // TODO: handle exception
        Log.e("Json parser", "error parsing data" +
e.toString());
    }
    return jsonObj;
}
}
```