



**APLIKASI PENJUALAN ONLINE PADA SANG SURYA SERBA MURAH
BATUSANGKAR DENGAN FRAMEWORK CODEIGNITER
BERBASIS WEB**

TUGAS AKHIR

*Ditulis Sebagai Syarat untuk Memperoleh Gelar Ahli Madya
(D.III)
Jurusan Manajemen Informatika*

Oleh:

**WIDIA NURMISIA
NIM : 1750401069**

**JURUSAN MANAJEMEN INFORMATIKA
FAKULTAS EKONOMI DAN BISNIS ISLAM
INSTITUT AGAMA ISLAM NEGERI
BATUSANGKAR
2021**

SURAT PERNYATAAN KEASLIAN

Saya yang bernama nama di bawah ini :

Nama : Widi Nurhidia
NIM : 175161099
Tempat/Tgl Lahir : Banjarmasin, 21 April 1999
Fakultas : Ekonomi dan Bisnis Islam
Jurusan : Manajemen Islamatika

Menyatakan dengan sesungguhnya bahwa Tugas Akhir saya yang berjudul "APLIKASI PENJUALAN ONLINE PADA KANG RUDYA BERBA MUHAPE BATHASANGKAR DENGAN FRAMEWORK CONSOLEWEB BERBASIS WEB" adalah benar-benar karya sendiri tidak plagiat dan tidak diuraikan ke orang lain.

Apabila dikemudian hari terbukti bahwa karya tersebut plagiat, maka saya bersedia menerima sanksi sesuai dengan peraturan - undang yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya, untuk digunakan sebagaimana mestinya.

Banjaringkar, 12 Agustus 2021
Saya yang menandatangani,



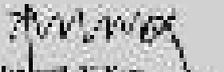
Widi Nurhidia
NIM. 175161099

PERSEMBAHAN PERSEMBAHAN

Persembahkan kepada: **Anggota Komisi Nasional Wanita Muhammadiyah (KNW) (2014-2016)**, dengan judul: **"Aplikasi Perjanjian Kerja yang Sesuai Syarat Mutual dan Sengketa Dengan Framework Kebijakan Buruh dan WPK"**, merupakan salah satu upaya dalam meningkatkan literasi hukum perburuh dan dapat diartikan sebagai **dituangkan dalam** **Keputusan Manajemen**.

Keputusan penyusunan dan pembahasan dapat dipergunakan sebagai referensi.

Ketua Komite Manajemen
Inisiatif:


Dedi N. Kera
NIP. 19740310 200712 1001

Direktur Jenderal DGT
Demokratik


Ellysa Hana, M.Hum
NIP.

Direktur
Buku Buruh, Perempuan dan Anak Buruh
Buruh Perempuan dan Anak Buruh (BPA),
Demokratik


Ellysa Hana, M.Hum
NIP. 19740310 200712 1001

ABSTRAK

Pokok permasalahan dalam Tugas Akhir ini adalah Pembuatan Aplikasi Penjualan pada toko. Tujuan Pembahasan ini untuk mempermudah penjual dan pembeli bertransaksi, meningkatkan omset penjualan Sang Surya Serba Murah Batusangkar dan dapat mempermudah mempromosikan produk-produk yang ada pada toko tersebut. Selain itu juga mempermudah penjual dalam pencatatan laporan penjualan.

Jenis penelitian yang digunakan adalah jenis penelitian lapangan (*Field Reseach*). Untuk mendapatkan data-data dari permasalahan yang diteliti. Teknik pengumpulan data yang penulis gunakan adalah melalui wawancara. Pengolahan data dilakukan secara deskriptif kualitatif, kemudian diuraikan serta melakukan klasifikasi terhadap aspek masalah tertentu dan memaparkan melalui kalimat yang efektif.

Dari penelitian yang dilakukan dapat disimpulkan bahwa penjualan pada toko ini dilakukan secara langsung sehingga penjualan terjadi hanya dalam kota saja yang mengakibatkan rendahnya pendapatan. Karena itu penulis bermaksud melakukan pembuatan aplikasi penjualan ini untuk menambah omset penjualan toko karena dapat bertransaksi secara luas. Dengan memanfaatkan aplikasi penjualan diharapkan dapat membantu proses transaksi jual-beli dan dengan memanfaatkan framework codeigniter berbasis web sebagai konsep dasar aplikasi diharapkan dapat menggantikan cara yang kurang efektif dan efisien serta diharapkan dapat mempermudah pembuatan laporan hasil proses jual-beli.

Kata Kunci : Aplikasi, Penjualan, Framework Codeigniter, Web

KATA PENGANTAR



Puji syukur penulis haturkan atas kehadiran Allah SWT. yang telah melimpahkan rahmat dan karunia-Nya sehingga pembuatan tugas akhir ini dapat terselesaikan. Salawat dan salam buat junjungan umat, yakni Nabi Muhammad SAW, yang telah membawa manusia dari alam kebodohan ke alam yang penuh dengan ilmu pengetahuan seperti kita rasakan seperti sekarang ini.

Dalam penulisan tugas akhir ini penulis banyak mendapat bimbingan dan bantuan baik moril maupun materil dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Marjoni Imamora, M.Sc selaku Rektor IAIN Batusangkar.
2. Bapak Dr. H. Rizal, M.Ag.,CRP selaku Dekan Fakultas Ekonomi dan Bisnis Islam.
3. Bapak Iswandi, M.Kom selaku ketua Jurusan Manajemen Informatika IAIN Batusangkar beserta jajarannya yang telah memberikan fasilitas dan layanan dalam proses perkuliahan dan penyelesaiannya.
4. Ibuk Lidya Rahmi, M.Pd.T selaku dosen pembimbing yang telah membimbing dan mengarahkan penulis selama penyusunan Tugas Akhir.
5. Bapak dan Ibuk Dosen yang telah memberikan ilmu kepada penulis selama masa perkuliahan.
6. Pihak Sang Surya Serba Murah Batusangkar yang telah banyak meluangkan waktunya untuk membantu penulis saat penelitian, dan selalu memberikan sambutan yang ramah.
7. Orang Tua dan keluarga tercinta yang telah banyak memberikan dorongan, semangat serta bantuan moril dan materil.

8. Kepada Sanak famili yang telah banyak memberikan semangat dan dorongan kepada penulis hingga penulis bisa menyelesaikan tugas akhir ini.
9. Teman-teman seperjuangan angkatan (2017) yang telah memberikan saran-saran yang bermanfaat dalam menyelesaikan tugas akhir ini.
10. Pihak-pihak yang tidak dapat penulis sebutkan satu persatu, yang telah memberikan motivasi dan semangat serta sumbangan pemikirannya kepada penulis sehingga selesainya tugas akhir ini.

Penulis sadar bahwasanya tugas akhir ini jauh dari kesempurnaan, oleh karena itu kritikan dan saran yang sifatnya membangun sangat penulis harapkan untuk kesempurnaan tugas akhir ini. Penulis juga berharap semoga penulisan tugas akhir ini memberikan manfaat kepada kita semua. Amin...

Akhirnya kepada Allah SWT jualah penulis bermohon dan bersujud semoga keikhlasan yang diberikan akan dibalas-Nya. *Amin Ya Robbal'alamin*.

Batusangkar, Agustus 2021

Penulis

DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	vii
DAFTAR GAMBAR.....	viii
BAB I PENDAHULUAN	1
A. Latar Belakang.....	1
B. Identifikasi Masalah	2
C. Batasan Masalah	2
D. Rumusan Masalah.....	2
E. Tujuan Penelitian.....	3
F. Manfaat Penelitian.....	3
G. Metode Penelitian	3
H. Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
A. Gambaran Umum Toko	5
1. Sejarah Toko	5
2. Visi Toko	5
a. Unggul Dalam Segi Harga.....	5
b. Menghasilkan jenis-jenis fashion berkualitas unggul.....	5
c. Terkemuka dan terdepan dalam pelayanan dan kinerja.....	5
d. Menjadi tujuan belanja di berbagai kalangan.	5
3. Misi Toko.....	5
4. Tujuan Toko.....	5
5. Struktur Organisasi Toko.....	6
6. Fungsi dan tugas	6
B. Konsep Dasar Aplikasi Penjualan	6
1. Aplikasi	6
2. <i>E-commerce</i>	8
3. Framework	10

4. Codeigniter.....	11
5. Website	11
6. HTML	11
7. MySql.....	11
8. PHP	15
C. Perangkat Lunak Pembangun Sistem	16
1. Xampp.....	16
2. Sublime Text.....	17
D. Alat Bantu Perancangan	18
1. Use Case Diagram	19
2. Activity Diagram	20
3. SequenceDiagram	21
4. Class Diagram.....	22
5. Desain Database.....	23
6. Desain Output	23
7. Desain Input.....	23
BAB III ANALISA DAN HASIL.....	24
A. Analisa Sistem	24
B. Perancangan Sistem.....	25
1. Peran <i>Actor</i>	25
2. <i>Use case diagram</i>	25
3. <i>Activity Diagram</i>	26
4. Sequence Diagram.....	29
5. Collaboration diagram	31
6. Class Diagram.....	32
7. Struktur Program.....	33
C. Desain Perancangan Terinci	33
1. Desain Output	34
2. Desain Input.....	36
3. Desain File	38
BAB IV PENUTUP	41

A. Kesimpulan.....	41
B. Saran	42

DAFTAR PUSTAKA

DAFTAR TABEL

Tabel 2. 1 Simbol-Simbol Use Case Diagram	19
Tabel 2. 2 Simbol-Simbol Activity Diagram	20
Tabel 2. 3 Simbol-Simbol Sequence Diagram	21
Tabel 2. 4 Simbol-simbol Class Diagram	22
Tabel 3. 1 Peran <i>Actor</i>	25
Tabel 3. 2 User	38
Tabel 3. 3 Barang	38
Tabel 3. 4 Transaksi	39
Tabel 3. 5 Konfirmasi	39

DAFTAR GAMBAR

Gambar 2. 1 Struktur organisasi toko Sang Surya	6
Gambar 3. 1 Use Case Diagram	26
Gambar 3. 2 Activity Diagram Admin.....	27
Gambar 3. 3 Activity Diagram Customer	28
Gambar 3. 4 Sequence Diagram Admin	29
Gambar 3. 5 Sequence Diagram Customer	30
Gambar 3. 6 Collaboration Diagram Admin.....	31
Gambar 3. 7 Collaboration Diagram Customer	31
Gambar 3. 8 Class Diagram	32
Gambar 3. 9 Struktur Program	33
Gambar 3. 10 Daftar Produk	34
Gambar 3. 11 Data Barang.....	34
Gambar 3. 12 Keranjang Belanja	35
Gambar 3. 13 Data Pemesanan Produk.....	35
Gambar 3. 14 Laporan Penjualan.....	36
Gambar 3. 15 Login	36
Gambar 3. 16 Tambah Data Barang.....	37
Gambar 3. 17 Alamat Pengiriman dan Pembayaran	37

BAB I

PENDAHULUAN

A. Latar Belakang

Teknologi informasi sangat berkembang pesat dan memberi berbagai manfaat bagi penggunanya. Karena semakin berkembangnya zaman semakin tinggi juga kecepatan teknologi dan semakin mudah diakses atau dipergunakan. Sehingga mendorong percepatan di berbagai bidang. Hal tersebut juga yang menyebabkan munculnya kemajuan pada perangkat lunak dan diimbangi pula dengan kemajuan dan kecanggihan teknologi beserta perangkat kerasnya. Secara langsung atau tidak langsung, teknologi informasi telah menjadi bagian penting dari berbagai kehidupan. Karena banyak kemudahan yang ditawarkan, teknologi informasi hampir tidak dapat di lepaskan dari berbagai aspek kehidupan manusia. (Elizaandayni,2013)

Sang Surya Serba Murah Batusangkar merupakan perusahaan yang bergerak dibidang penjualan pakaian. Dengan model-model baru yang di update setiap saat. Sistem penjualan yang digunakan Sang Surya Serba Murah Batusangkar saat ini masih menggunakan cara manual dengan datang langsung ke toko, karena belum memiliki aplikasi penjualan online (e-commerce) sehingga penjualan masih minim. Ditambah lagi dengan situasi pandemi saat ini, adanya pembatasan sosial pun menyebabkan berbagai aktivitas mesti dilakukan jarak jauh. Sehingga semua aktivitas dilakukan dirumah. Mengakibatkan penurunan penjualan dan pendapatan sehingga banyak barang yang tidak terjual dan menumpuk di gudang.

Solusi yang diajukan untuk mengatasi masalah pada Sang Surya Serba Murah Batusangkar untuk mengembangkan bisnis penjualannya yaitu aplikasi penjualan (e-commerce). Aplikasi penjualan merupakan suatu kontak transaksi perdagangan antara penjual dan pembeli dengan menggunakan media internet. Keuntungan yang diperoleh dengan menggunakan transaksi melalui e-commerce yaitu untuk meningkatkan pendapatan dengan menggunakan penjualan online yang efektif dan efisien.

Berdasarkan uraian diatas penulis menyimpulkan dengan adanya aplikasi penjualan dapat mempermudah transaksi kapan saja dan dimana saja. Sehingga penulis tertarik untuk melakukan penelitian dan mengangkat permasalahan ini kedalam sebuah tugas akhir dengan judul : “ **Aplikasi Penjualan Online Pada Sang Surya Serba Murah Batusangkar Dengan Framework Codeigniter Berbasis WEB**“.

B. Identifikasi Masalah

Berdasarkan latar belakang diatas, maka penulis dapat mengidentifikasi beberapa masalah yaitu :

1. Dalam melakukan pembelian pakaian pada Sang Surya Serba Murah Batusangkar konsumen datang langsung ketoko.
2. Belum ada informasi mengenai produk terbaru dan ketersediaan pakaian, ukuran, warna, harga, bahan Sang Surya Serba Murah Batusangkar.
3. Promosi yang kurang luas menyebabkan jangkauan pembeli masih lingkup yang kecil dan terbatas.
4. Sistem laporan penjualan masih dilakukan secara manual.

C. Batasan Masalah

Berdasarkan identifikasi masalah diatas dan mengingat keterbatasan waktu, dan biaya, maka dalam hal ini penulis membatasi masalah, hanya akan membahas mengenai aplikasi penjualan pakaian pada Sang Surya Serba Murah Batusangkar.

D. Rumusan Masalah

Berdasarkan latar belakang masalah yang dikemukakan diatas, maka dapat dirumuskan suatu masalah yang akan dibahas, yaitu:

1. bagaimana merancang aplikasi penjualan online dengan framework codeigniter berbasis web di Sang Surya Seba Murah Batusangkar yang mampu mengelola informasi produk, trasaksi dan laporan penjualan?
2. bagaimana cara menerapkan penggunaan aplikasi penjualan online di Sang Surya Seba Murah Batusangkar?

E. Tujuan Penelitian

Dalam penulisan laporan ini memiliki tujuan yang akan dicapai, adapun tujuan tersebut adalah :

1. Memberikan kemudahan kepada konsumen mendapatkan informasi dalam membeli produk yang di jual oleh Sang Surya Serba Murah Batusangkar.
2. Membantu Sang Surya Serba Murah Batusangkar dalam mempromosikan dan menjual produk kepada konsumen.
3. Merancang sistem dalam pengolahan data transaksi serta pembuatan laporan.

F. Manfaat Penelitian

1. Sebagai implementasi dan pengembangan ilmu yang telah penulis dapatkan selama masa perkuliahan.
2. Sebagai syarat untuk memperoleh gelar Ahli Madya Program Diploma III (D.III) Manajemen Informatika IAIN Batusangkar.
3. Dapat memberikan sumbangan pemikiran pada Sang Surya Serba Murah Batusangkar agar memetik manfaat dari teknologi informasi.

G. Metode Penelitian

Dalam penulisan ini, penulis menggunakan beberapa metode antarlain :

1. Penelitian lapangan

Data yang diperoleh dalam penelitian lapangan ini untuk melihat kenyataan yang sebenarnya dalam masalah yang diteliti dengan cara sebagai berikut:

- a. Observasi

Teknik pengumpulan data dengan mengadakan penelitian dan peninjauan langsung ke Sang Surya Serba Murah Batusangkar

- b. Interview

Dalam penelitian ini, wawancara dilakukan dengan tujuan untuk mengidentifikasi dan menganalisa kebutuhan perangkat lunak yang akan dikembangkan. Wawancara dilakukan dengan pemilik toko,

karyawan atau pihak terkait tentang masalah yang menjadi objek pengamatan atau pembahasan.

2. Penelitian Kepustakaan

Penelitian ini dilakukan untuk mencari, mengumpulkan dan mempelajari data dari buku-buku, bahan kuliah, karangan ilmiah, ataupun tulisan yang berhubungan dengan permasalahan yang dibahas dalam penelitian ini.

3. Penelitian Laboratorium

Penulis melakukan pengolahan data yang telah dikumpulkan dengan menggunakan komputer sebagai alat bantu pembuatan Tugas Akhir ini.

H. Sistematika Penulisan

Penulisan ini dilakukan dengan membagi penjelasannya pada beberapa BAB, dimana tahap dan struktur dari penulisannya dapat dilihat sebagai berikut :

1. BAB I yang merupakan pendahuluan yang terdiri dari: Latar Belakang, Identifikasi Masalah, Batasan Masalah, Rumusan Masalah, Tujuan Penelitian, Manfaat Penelitian, Metodologi Penelitian dan Sistematika Penulisan.
2. BAB II yang menguraikan tentang teori yang berhubungan dengan judul Tugas Akhir, seperti mengenai sejarah toko, Struktur organisasi, visi dan misi, konsep dasar aplikasi dan perancangan system dan sekilas tentang framework codeigniter.
3. BAB III merupakan Analisa dan Hasil, terdiri dari Analisa Sistem dan Perancangan system yang baru.
4. BAB IV terdiri dari kesimpulan dan saran, Bab ini berisikan kesimpulan seluruh rangkaian kegiatan selama proses penelitian yang menyatakan kelebihan dan kekurangan dari hasil selama penelitian berlangsung, serta saran-saran bagi perbaikan dan pengembangan sistem yang dimungkinkan.

BAB II

LANDASAN TEORI

A. Gambaran Umum Toko

1. Sejarah Toko

Sang Surya Serba Murah Batusangkar merupakan salah satu toko fashion yang sudah cukup lama berdiri di Batusangkar. Toko ini awalnya sebuah toko yang hanya menjual baju wanita. Seiring berjalannya waktu toko ini mengalami perkembangan dan penjualan yang cukup baik sehingga pemilik berinisiatif menambah ragam penjualan seperti dengan menjual pakaian pria, pakaian anak dan juga hijab. Didirikan oleh Edwin Rafil pada November 2016 yang beralamat di Jl. Ahmad Yani No.89 A – 89 B – 89 C – 89 D Sigarunggung Batusangkar.

2. Visi Toko

- a. Unggul Dalam Segi Harga.
- b. Menghasilkan jenis-jenis fashion berkualitas unggul.
- c. Terkemuka dan terdepan dalam pelayanan dan kinerja.
- d. Menjadi tujuan belanja di berbagai kalangan.

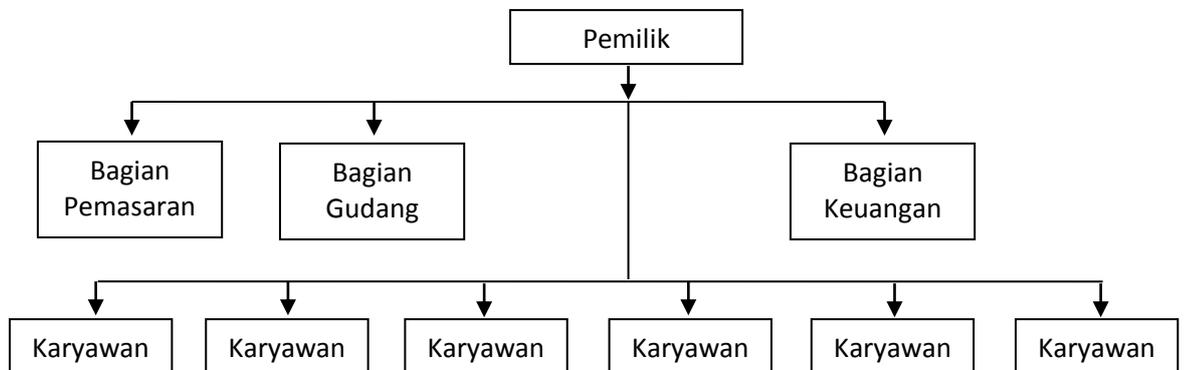
3. Misi Toko

- a. Menyediakan berbagai macam fashion wanita, pria dan anak-anak dengan harga yang jauh lebih murah dibanding toko lainnya yang sejenis.
- b. Mampu menyediakan variasi pilihan baju yang selalu mengikuti trend saat ini.
- c. Memberikan pelayanan terbaik demi menjalin hubungan jual beli yang baik dengan konsumen serta melayani dengan setulus hati.

4. Tujuan Toko

- a. Menjual fashion wanita, pria dan anak-anak dengan harga yang relatif lebih murah.
- b. Menjadi toko fashion yang disenangi konsumen karena pelayanan baik

5. Struktur Organisasi Toko



Gambar 2. 1
Struktur organisasi toko Sang Surya

6. Fungsi dan tugas

a. Pemilik

Pemilik bertugas sebagai pengatur berjalannya mekanisme usaha

b. Bagian Pemasaran

Bagian pemasaran bertugas memasarkan produk

c. Bagian Gudang

Bagian gudang bertugas melakukan pencatatan keluar masuk nya barang

d. Bagian Keuangan

Bagian keuangan bertugas sebagai pengelola uang masuk dan uang keluar toko

e. Karyawan

Karyawan bertugas melayani pembeli

B. Konsep Dasar Aplikasi Penjualan

1. Aplikasi

a. Pengertian Aplikasi

Aplikasi adalah suatu sub kelas perangkat lunak komputer yang memanfaatkan kemampuan komputer berlangsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan

komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna.

Pengertian aplikasi menurut Barry Pratama:

- 1) Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas.
- 2) Aplikasi adalah sistem lengkap yang mengerjakan tugas spesifik.
- 3) Aplikasi basisdata terdiri atas sekumpulan menu, formulir, laporan dan program yang memenuhi kebutuhan suatu fungsional unit bisnis/ organisasi/ instansi.

Menurut Ibis, aplikasi adalah alat bantu untuk mempermudah dan mempercepat proses pekerjaan dan bukan merupakan beban bagi penggunanya. Beberapa aplikasi yang digabung bersama menjadi suatu paket disebut sebagai suatu paket atau application suite. Aplikasi dalam satu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Software application adalah software program yang memiliki aktivitas pemrosesan perintah yang diperlukan untuk melaksanakan permintaan pengguna dengan tujuan tertentu. Software application terdiri dari bahasa pemrograman (programming language), program aplikasi (application program), program paket atau paket aplikasi (package program), program utilitas (utility program), games, entertainment, dan lain-lain. Untuk mendukung operasi software application dia atas, tugas pengguna komputer dibagi menjadi beberapa bagian yaitu sebagai Analisis Sistem, Programmer, Operator, Administrator Database, Administrator jaringan.

b. Komponen Aplikasi

Aplikasi terdiri dari komponen-komponen sebagai berikut:

- 1) Komponen *Input* (masukan)

Input merupakan data yang masuk ke dalam aplikasi. Meliputi metode-metode dan media untuk menangkap data yang akan dimasukkan, dapat berupa dokumen-dokumen dasar.

2) Komponen *Output* (keluaran)

Output adalah informasi yang berkualitas dan dokumentasi yang berguna *untuk* semua tingkatan manajemen serta semua pemakai sistem. Dapat berupa keluaran dokumen dan informasi yang berkualitas.

3) Komponen Teknologi

Teknologi merupakan alat dalam aplikasi. Teknologi digunakan untuk *menerima input*, menjalankan model, penyimpan dan mengakses data, menghasilkan dan mengirimka *output*, dan membantu pengendalian sistem.

4) Blok Basis Data

Merupakan kumpulan data yang berhubungan satu dengan yang lainnya, tersimpan diperangkat keras computer dan perangkat lunak dengan menggunakan *software database*.

5) Blok Kendali

Pengendalian yang dirancang untuk mengurangi gangguan terhadap aplikasi.

2. *E-commerce*

a. Pengertian *E-commerce*

Electronic commerce (e-commerce) menurut nugroho(2006:5) perdagangan elektronik didefinisikan sebagai “cara untuk menjual dan membeli barang-barang dan jasa lewat jaringan internet, tetapi hal ini mencakup berbagai hal aspek diantaranya transaksi pembelian serta tranfer dana via jaringan komputer”. *E-commerce* merupakan sekumpulan teknologi beserta aplikasinya yang menghubungkan perusahaan, konsumen(consumers), manufaktur, internet service provider (ISP) dan pedagang perantara (intermediateries) melalui transaksi-transaksi elektronik dan pertukaran melalui elektronik dalam bentuk barang, jasa dan informasi.

b. Jenis E-commerce

- 1) Business to business (B2B), yaitu kegiatan bisnis yang terjadi antar perusahaan atau produsen;
- 2) Business to consumer (B2C), yang terjadi pada pelanggan, perusahaan penjual jasa dan perusahaan retail online;
- 3) Consumer to business (C2B), yaitu kegiatan bisnis yang terjadi antara konsumen dan produsen;
- 4) Government to business (G2B), yaitu kegiatan bisnis yang terjadi di antara pemerintah dan pengusaha;
- 5) Government to consumer (G2C), yaitu kegiatan bisnis yang terjadi di antara pemerintah dan konsumen.

c. Kategori E-commerce

1) Electronic Markets

Electronic Markets merupakan penggunaan teknologi informasi dan komunikasi untuk menampilkan berbagai penawaran yang tersedia dalam segmen pasar, sehingga pembeli dapat membandingkan harga (dan atribut lainnya) dari berbagai penawaran yang ada dan membuat keputusan. Contoh umum dari electronic markets ialah sistem pemesanan pesawat terbang.

2) EDI (Electronic Data Interchange)

EDI menyediakan sistem standar untuk pengkodean transaksi perdagangan sehingga mereka dapat dikomunikasikan secara langsung dari satu sistem computer ke sistem computer lainnya tanpa membutuhkan perintah tertulis, tagihan, penundaan dan kesalahan dalam penanganan media kertas. EDI digunakan oleh organisasi yang memakai transaksi biasa dalam jumlah besar. Salah satu sector dimana EDI digunakan secara ekstensif adalah pertukaran di dalam supermarket yang menggunakan EDI untuk bertransaksi dengan supplier mereka.

3) Internet commerce

Teknologi komunikasi dan informasi juga dapat digunakan untuk periklanan dan membuat satu kali penjualan dalam daerah produk jasa yang besar. E-commerce jenis ini dilambungkan dengan pemakaian internet secara komersial. Internet dapat digunakan untuk pembelian buku yang dapat dikirim melalui pos atau pemesanan tiket yang dapat dilakukan oleh klien kapan saja mereka inginkan. Hal ini dapat ditandai dengan internet bukan satu-satunya teknologi yang digunakan untuk pelayanan jenis ini dan ini bukan satu-satunya kegunaan internet dalam e-commerce.

d. Keuntungan E-commerce

- 1) Bagi konsumen : Harga lebih murah, belanja cukup pada satu tempat
- 2) Bagi pengelola bisnis : efisiensi, tanpa kesalahan, tepat waktu
- 3) Bagi manajemen : peningkatan pendapatan, loyalitas pelanggan

e. Kerugian E-commerce

- 1) Persoalan tidak bias di selesaikan seperti nama domain, copyright, pajak dan biaya-biaya lainnya.
- 2) Kurangnya peraturan dari pemerintah nasional maupun internasional dan standar industry.
- 3) Banyak pembeli dan penjual yang menunggu e-commerce itu stabil sebelum mereka ikut berpartisipasi.
- 4) Persepsi bahwa e-commerce itu mahal dan tidak terjamin keamanannya.

3. Framework

Menurut Jubilee (2015) *framework* merupakan banyak kode, yang disimpan kedalam beberapa file yang terpisah, dan memberikan kemudahan penggunaan kode yang digunakan secara berulang-ulang. Kumpulan fungsi dari kumpulan *library*. Sehingga seorang *programmer* tidak perlu lagi membuat fungsi-fungsi dari awal. Karena sudah dapat memanggil fungsi-fungsi yang telah ada, tentunya cara menggunakan

fungsi-fungsi tersebut sudah ditemukan oleh *framework*. (Bernadus & Ester, 2012:1)

4. Codeigniter

Menurut Naista (2017) codeigniter merupakan *framework* berbasis PHP yang kuat dengan *footprint* yang sangat kecil, dibangun untuk pengembang yang membutuhkan *toolkit* sederhana dan elegan untuk membuat aplikasi web dengan fitur lengkap. Codeigniter sendiri yaitu merupakan *framework* PHP yang memiliki model MVC (Model, View, Controller) atau memiliki *tools* berupa beberapa folder untuk mengcoding dan memanggil beberapa fungsi dalam folder tersebut untuk memberikan kemudahan membuat website.

5. Website

Menurut Rahmad (2010) *website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman web dengan halaman lain disebut Hyperlink, sedangkan teks yang dijadikan media penghubung disebut Hypertext.

6. HTML

Menurut Anhar (2010) *Hypertext Markup Language* (HTML) merupakan bahasa pemrograman web yang memiliki sintak atau aturan tertentu dalam menuliskan *script* atau kode-kode, sehingga browser dapat menampilkan informasi dengan membaca kode-kode HTML. Untuk memudahkan kita dalam mempelajari dasar pemrograman HTML.

7. MySql

Menurut Sugiri (2008: 1), MySQL merupakan database yang bersifat *client server*, dimana data diletakkan di *server* yang bisa diakses melalui komputer *client*. MySQL dibuat sekitar tahun 1994/1995 dan dikembangkan oleh sebuah perusahaan di Swedia yang bernama MySQL

AB dengan istilah T.c.X DataKonsult AB. Tujuannya adalah untuk mengembangkan aplikasi web yang dimiliki oleh kliennya.

Menurut Andi (2011) Database atau sering disebut basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan merupakan sumber sistem informasi yang dapat diperiksa menggunakan suatu program komputer. *Database* berfungsi untuk menyimpan informasi atau data. Database merupakan sekumpulan dan sebagai penyediaan informasi bagi pengguna atau *user*. Untuk mengolah database diperlukan *software* yang sering disebut dengan DBMS (*Database Management System*). Dengan DBMS pengguna atau user dapat membuat, mengelola, mengontrol, dan mengakses database dengan mudah, praktis dan efisien.

Database terdiri dari table yang didalamnya *terdapat field-field*, dan sebuah database bias terdiri dari beberapa table. Dalam pembuatan database, hal-hal yang perlu diperhatikan adalah sebagai berikut

- a. Setiap table dalam *database*, harus memiliki *field* (kolom) yang unik disebut dengan *primary key*.
- b. Table dalam *database* tidak boleh ada *redundancy* data yaitu mengandung *record* ganda. Jika terdapat data yang sama, maka perlu dilihat kembali rancangan tabelnya.
- c. Pilih tipe data yang tepat, sehingga ukuran *database* seminimal mungkin.

MySQL merupakan *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structure Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan/seleksi dan pemasukkan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Kendala

suatu sistem database (DBMS) dapat diketahui dan cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai *database server*, *MySQL* dapat dikatakan lebih unggul dibandingkan dengan *database server* yang lainnya dalam *query* data.

a. Keistimewaan MySQL

Sebagai *database* yang dimiliki konsep *database* modern, *MySQL* memiliki banyak sekali keistimewaan yang dimiliki oleh *MySQL*

1) *Portability*

MySQL dapat berjalan stabil pada berbagai sistem operasi di antaranya adalah seperti *Windows*, *Linux*, *FreeBSD*, *Mac OS X server*, *Solaris*, *Amiga*, *HP-UX* dan masih banyak lagi.

2) *Open Soucre*

MySQL didistribusikan secara *open soure (gratis)*, di bawah lisensi *GPL*.

3) *Multiuser*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang *bersamaan* tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah *database server MySQL* dapat diakses *client* secara bersamaan.

4) *Performance Tuning*

MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana dengan kata lain dapat memproses lebih banyak *SQL* per satuan waktu.

5) *Column Types*

MySQL memiliki tipe kolom yang sangat kompleks, seperti *signed/ unsigned integer*, *fload*, *double*, *char*, *varchar*, *text*, *blob*, *date*, *time*, *datetime*, *year*, serta *enum*.

6) *Command dan Function*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *SELECT* dan *WHERE* dalam *query*.

7) *Security*

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama *host*, dan *user* dengan *system* perizinan yang mendetail serta password terensripsi.

8) *Stability dan Limits*

MySQL mampu menangani *database* dalam skala besar, dengan jumlah *records* lebih dari 50 juta dan 60 ribu *table* serta 5 miliar baris. Selain itu, batas *indeks* yang dapat ditampung mencapai 32 *indeks* pada tiap tabelnya.

9) *Connectivity*

MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol TCP/IP*, *Unix socket (Unix)*, atau *Named Pipes (NT)*.

10) *Localisation*

MySQL dapat mendeteksi pesan kesalahan (*error code*) pada *client* dengan menggunakan lebih dari dua puluh bahasa. Meski demikian, bahasa Indonesia belum termasuk di dalamnya.

11) *Interface*

MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12) *Client dan Tools*

MySQL dilengkapi dengan berbagai *tool* yang dapat digunakan untuk administrasi *database*, dan pada setiap *tool* yang ada disertai petunjuk *online*.

13) *Struktur Tabel*

MySQL memiliki *struktur table* yang lebih fleksibel dalam menangani *ALTER TABLE*, dibanding *database* lainnya semacam *PostgreSQL* ataupun *Oracle*.

8. PHP

Menurut Supomo dan Vidiandry (2016) *HyperText Preprocessor* adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambah kedalam HTML. PHP merupakan suatu bahasa pemrograman sisi server yang dapat anda gunakan untuk membuat halaman Web dinamis. Contoh bahasa yang lain adalah *Microsoft Active Server Page* (ASP) dan *Java Server Page* (JSP). Dalam suatu halaman HTML anda dapat menanamkan kode PHP yang akan dieksekusi setiap kali halaman tersebut dikunjungi. Karena kekayaannya akan fitur yang mempermudah perancangan dan pemrograman web, PHP memiliki popularitas yang tinggi. PHP adalah kependekan dari *HyperText Preprocessor* (suatu akronim rekursif) yang dibangun oleh Rasmus Lerdorf pada tahun 1994. Dahulu, pada awal perkembangannya PHP disebut sebagai kependekan dari *Personal Home Page*. PHP merupakan produk *Open source* sehingga dapat mengakses *source code*, menggunakan, mengubahnya tanpa harus membayar sepersenpun. (Antonius, 2010:9).

a. Sejarah Singkat PHP

PHP diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *homepage*-nya. Pada tahun 1996, PHP telah banyak digunakan dalam website di dunia.

b. Kelebihan-Kelebihan PHP

PHP memiliki banyak kelebihan yang tidak dimiliki oleh bahasa script sejenis. Kelebihan-kelebihan diantaranya adalah :

- 1) PHP difokuskan pada pembuatan *script server-side*, yang bias melakukan apa saja yang dapat dilakukan oleh CGI, seperti mengumpulkan data dari *form*, menghasilkan isi halaman *web* dinamis, dan kemampuan mengirim serta menerima *cookies*, bahkan lebih dari pada kemampuan CGI.

- 2) PHP dapat digunakan pada semua sistem operasi antara lain *linux*, *Unix* (termasuk variasinya *HP-UX*, *Solaris* dan *OpenBSD*), *Microsoft windows*, *Mac OS X*, *RISC OS*.
- 3) PHP mendukung banyak *WEB Server* seperti *Apache*, *Microsoft Internet Information Server (MIS)*, *Personal Web Server (PWS)*,
- 4) dan masih banyak lagi lainnya, bahkan PHP dapat bekerja sebagai suatu *CGI processor*.
- 5) PHP tidak terbatas pada hasil keluaran HTML. (*Hypertext Markup Languages*). PHP juga memiliki kemampuan untuk mengolah keluaran gambar, *File PDF*, dan *Movies Flash*. PHP juga dapat menghasilkan teks seperti *XHTML* dan *file XML* lainnya.

c. Sintak PHP

Sintak program / *script* PHP ditulis dalam apitan tanda khusus PHP. Ada empat macam pasangan tag PHP yang dapat digunakan untuk menandai blok script PHP :

- 1) `<?php...?>`
- 2) `<script language = "PHP">...</script>`
- 3) `<? ... ?>`
- 4) `<% ... %>`

Langkah Pertama merupakan format yang dianjurkan tetapimungkincara kedua akan sering digunakan karena lebih ringkas. Langkah ketiga digunakan untuk mengantisipasi editor-editor yang tidak dapat menerima kedua caradiatas.Langkah keempat juga dimungkinkan sebagai kemudahan bagi anda yang sudah terbiasa dengan *ASP (Active Server Pages)*. Namun, bila itu tidak dikenal, maka harus dilakukan pengaktifan pada *file* konfigurasi PHP ini.

C. Perangkat Lunak Pembangun Sistem

1. Xampp

Xampp adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan komplikasi dari beberapa program. Fungsinya

adalah sebagai server yang berdiri sendiri (localhost), yang terdiri dari atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem apapun), Apache, MySQL, 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.

2. Sublime Text

Sublime Text adalah editor kode sumber lintas platform yang eksklusif dengan antarmuka pemrograman aplikasi *Python* (API). Ini secara native mendukung banyak bahasa pemrograman dan bahasa markup, dan fungsi dapat ditambahkan oleh pengguna dengan plugin, biasanya dibuat oleh komunitas dan dipelihara di bawah lisensi perangkat lunak bebas atau *freeware*.

Versi terakhirnya yaitu versi 3 memasuki yang memasuki tahap *beta* pada 29 Januari 2013. Awalnya hanya tersedia untuk pengguna terdaftar yang telah membeli Sublime Text versi sebelumnya, pada 28 Juni 2013 itu menjadi tersedia untuk masyarakat umum. Namun, pengembangan pengembangan terbaru masih membutuhkan kode registrasi. Sublime Text 3 secara resmi dirilis pada 13 September 2017. Dua fitur utama yang Sublime Text 3 menambahkan termasuk pengindeksan simbol dan manajemen panel.

Symbol Indexing memungkinkan Sublime Text untuk memindai file dan membangun indeks untuk memfasilitasi fitur *Goto Definition* dan *Goto Symbol in Project Management Pane* memungkinkan pengguna untuk berpindah antar panel melalui *hotkey*.

D. Alat Bantu Perancangan

Menurut Dharwiyanti (2003) Permodelan (*Modeling*) adalah proses merancang piranti lunak sebelum melakukan pengkodean (*coding*). Membuat model dari sebuah sistem yang kompleks sangat penting agar dapat memahami sistem secara menyeluruh. Semakin kompleks sebuah sistem, semakin penting pula penggunaan teknik pemodelan yang baik. Dengan menggunakan model, diharapkan pengembangan piranti lunak dapat memenuhi semua kebutuhan pengguna dengan lengkap dan tepat. Kesuksesan suatu pemodelan piranti lunak ditentukan oleh tiga unsur, yaitu pemodelan (*notation*), proses (*process*), dan *tool* yang digunakan.

Berdasarkan penjelasan Dharwiyanti (2003), penulis menggunakan perancangan sistem dengan pemodelan berorientasi objek menggunakan *Unified Modelling Language* (UML). Nugroho (2005) berpendapat bahwa UML, merupakan bahasa untuk visualisasi, spesifikasi, konstruksi, serta dokumentasi. Sependapat dengan, Dharwiyanti (2003) yang menjelaskan UML seperti sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

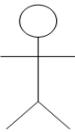
UML merupakan pemodelan berorientasi objek dalam merancang suatu sistem, akan tetapi dapat digunakan untuk pemodelan aplikasi prosedural. Pernyataan tersebut dikuatkan oleh Dharwiyanti (2003) dengan menggunakan UML dapat dibuat model untuk semua jenis aplikasi piranti lunak dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun serta ditulis dalam bahasa pemrograman apapun, karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. Menurut Nugroho (2005) Setiap sistem yang kompleks seharusnya bisa dipandang dari sudut yang berbeda-beda sehingga bisa didapatkan pemahaman secara menyeluruh. UML menyediakan sembilan jenis diagram yaitu *Diagram Class*, *Diagram Objek*, *Use Case*

Diagram, Sequence Diagram, Collaboration Diagram, Statechart Diagram, Activity Diagram, Component Diagram, Deployment Diagram. Akan tetapi Sulistyorini (2009) menyatakan bahwa kesembilan diagram tersebut tidak mutlak harus digunakan dalam pengembangan perangkat lunak, semua dibuat sesuai dengan kebutuhan.

1. Use Case Diagram

Use Case Diagram bersifat statis, diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna (Nugroho, 2005). Simbol-simbol yang digunakan dalam *Use Case Diagram* dijelaskan pada tabel berikut:

Tabel 2.1
Simbol-Simbol Use Case Diagram

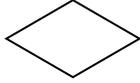
No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
2		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
3		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas
4		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>Independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>Independent</i>)
5		<i>Generalizat</i>	Hubungan dimana objek anak

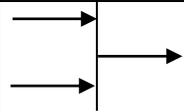
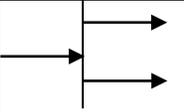
		<i>ion</i>	(<i>Descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>Ancestor</i>)
6		<i>InClude</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i>
7		<i>ExTend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan
8		<i>Assosiation</i>	Menghuungkan antara objek satu dengan objek lainnya

2. Activity Diagram

Grady Booch (2005) berpendapat bahwa, *An activity diagram is essentially a flowchart, showing flow of control from activity to activity*, *activity diagram* secara esensial mirip dengan *flowchart* atau diagram alur yang menunjukkan aliran kendali dari sebuah aktivitas ke aktivitas lainnya. Dalam *activity diagram* terdapat aksi atau aktivitas, *activity nodes*, *flows* atau aliran, dan objek. Simbol-simbol yang dipakai dalam *activity diagram* yaitu:

Tabel 2. 2
Simbol-Simbol Activity Diagram

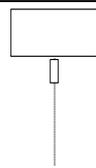
No	Simbol	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Decision</i>	Pilihan untuk pengambilan keputusan
3		<i>Initial Node</i>	Titik awal
4		<i>Actifity Final</i>	Titik akhir

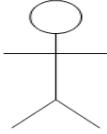
		<i>Node</i>	
5		<i>Join</i>	Penggabungan
6		<i>Fork</i>	Percabangan

3. SequenceDiagram

Sequence diagram menggambarkan interaksi antara objek di dalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu, menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan *output* tertentu. *Collaboration diagram* juga menggambarkan interaksi antara objek seperti *sequence diagram*, akan tetapi lebih menekankan pada masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, dimana *message* dari level tertinggi memiliki nomor 1 (Dharwiyanti, 2003). Simbol-simbol yang digunakan dalam *sequence diagram* dijelaskan pada tabel berikut:

Tabel 2. 3
Simbol-Simbol Sequence Diagram

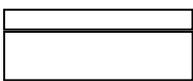
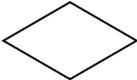
No	Simbol	Nama	Keterangan
1		<i>Object dan lifeline</i>	Orang, tempat, benda, kejadian atau konsep yang ada dalam dunia nyata yang penting bagi suatu aplikasi yang saling berinteraksi
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi

3		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i>
---	---	--------------	---

4. Class Diagram

Class adalah sebuah spesifikasi yang akan menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem. *Class diagram* menggambarkan struktur dan deskripsi kelas, package beserta hubungan satu sama lain (Dharwiyanti, 2003). Simbol-simbol yang digunakan dalam *class diagram* yaitu:

Tabel 2. 4
Simbol-simbol Class Diagram

No	Simbol	Nama	Keterangan
1		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		<i>Asosiasi</i>	Hubungan statis antar <i>class</i> yang menggambarkan <i>class</i> yang memiliki atribut berupa <i>class</i> lain atau <i>class</i> yang harus mengetahui eksistensi <i>class</i> lain
4		<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor)
5		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen

			mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (independent)
--	--	--	--

5. Desain Database

Nugroho (2005) menyatakan desain *database* adalah untuk mendefinisikan isi atau struktur dari tiap-tiap file yang telah diidentifikasi didesain secara umum. Elemen-elemen data disuatu file database harus dapat digunakan untuk pembuatan suatu output. Demikian juga dengan input yang direkamkan di *database*, file-file *database* harus mempunyai elemen-elemen untuk menampung input yang dimasukkan. Dengan demikian isi atau struktur dari suatu file database tergantung dari arus data masuk dan arus data keluar dari file tersebut.

6. Desain Output

Menurut Jogiyanto (2005) Output (keluaran) adalah produk dari sistem yang dapat dilihat. Istilah output ini terkadang dapat membingungkan, karena output dapat berupa hasil di media kertas (*microfilm*) atau hasil di media lunak (berupa tampilan layar video). Yang akan dimaksud output pada tahap desain ini adalah output yang berupa tampilan di media kertas atau di layar. Wahyono (2004) menyatakan output adalah salah satu tujuan dari dibuatnya sebuah program aplikasi.

7. Desain Input

Wahyono (2004) menyatakan Desain Input adalah desain interace form-form yang digunakan untuk memasuka data. Dan menurut Jogiyanto (2005) adalah bentuk dari dasar dokumen yang digunakan untuk menangkap data, dan bentuk dari tampilan input dari alat input.

BAB III

ANALISA DAN HASIL

A. Analisa Sistem

Analisis sistem merupakan tahapan paling awal dari pengembangan sistem yang menjadi fondasi menentukan keberhasilan sistem informasi yang dihasilkan nantinya. Tahapan ini sangat penting karena menentukan bentuk sistem yang nantinya akan dibangun. Seorang analis sistem harus memahami persoalan-persoalan yang ada sebelum menentukan tindakan apa saja yang akan dilakukan serta keputusan apa yang harus diambil dalam menyelesaikan persoalan tersebut. Disamping itu dengan menggambarkan setiap proses secara detail seorang analis harus mampu membuat rincian sistem dari hasil analisa menjadi bentuk perancangan sistem yang nantinya bisa di mengerti oleh pengguna sistem. Dalam perancangan sistem akan dijelaskan bagaimana bagian-bagian dari sistem informasi diimplementasikan.

Berikut ini adalah gambar proses alur sistem informasi sistem yang sedang ada Toko Sang Surya Serba Murah Batusangkar:

1. Pelanggan datang ke toko Sang Surya Serba murah untuk melihat barang yang diinginkan.
2. Setelah pelanggan menentukan barang yang ingin dibeli selanjutnya pihak toko melakukan pengecekan atas barang apakah barang yang diinginkan pelanggan tersedia.
3. Pihak Toko memberitahu pada pelanggan apakah barang yang diinginkan ada atau tidak.
4. Jika barang tersedia, Pihak toko sang surya serba murah memberitahu pada pelanggan berapa harga barang yang diinginkan.
5. Pelanggan melakukan pembayaran.
6. Pihak toko memberikan nota kepada pelanggan.
7. Pemilik toko membuat laporan penjualan.

B. Perancangan Sistem

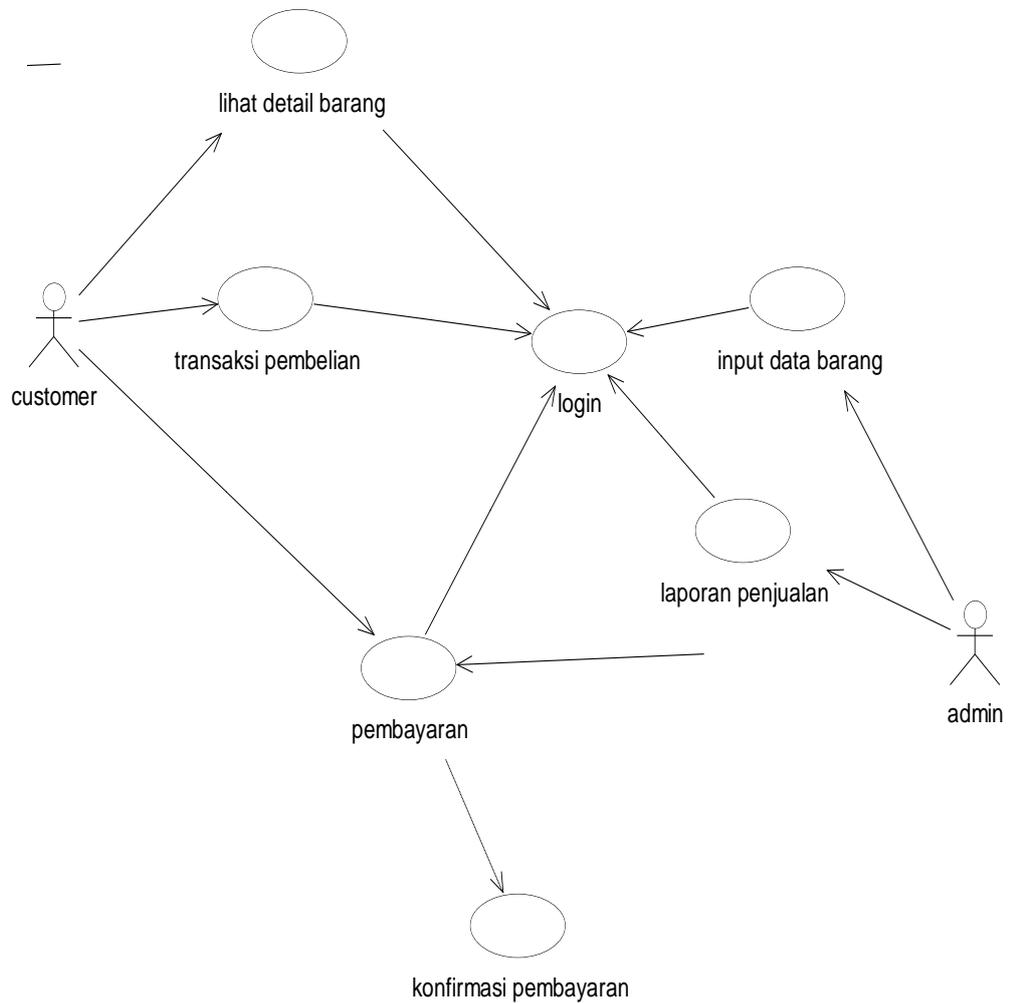
1. Peran Actor

Tabel 3. 1
Peran Actor

No	Actor	Peran
1	Admin	<ul style="list-style-type: none"> a. Login b. Menginputkan data barang c. Menambahkan data barang d. Mengedit data barang e. Menghapus data barang f. Admin mengelola data user g. Mencek konfirmasi pembayaran h. Admin melihat laporan barang. i. Mencetak Laporan penjualan j. Logout.
2	Customer	<ul style="list-style-type: none"> a. Login b. Customer melihat kategori barang c. Customer melakukan transaksi keranjang belanja. d. Costumer menampilkan transaksi belanja e. Mengkonfirmasi pembayaran f. Logout

2. Use case diagram

Use case diagram memperlihatkan suatu urutan interaksi antara aktor dan sistem. Seperti pada gambar berikut dimana *actor* (admin) melakukan login. Kemudian admin mengelola data user, admin menginputkan data kategori dan barang. Admin melihat laporan barang, pelanggan, pemesanan dan konfirmasi. Dan pada actor (customer) melakukan login kemudian customer melakukan transaksi keranjang belanja, dan menampilkan transaksi dan customer melihat kategori barang.

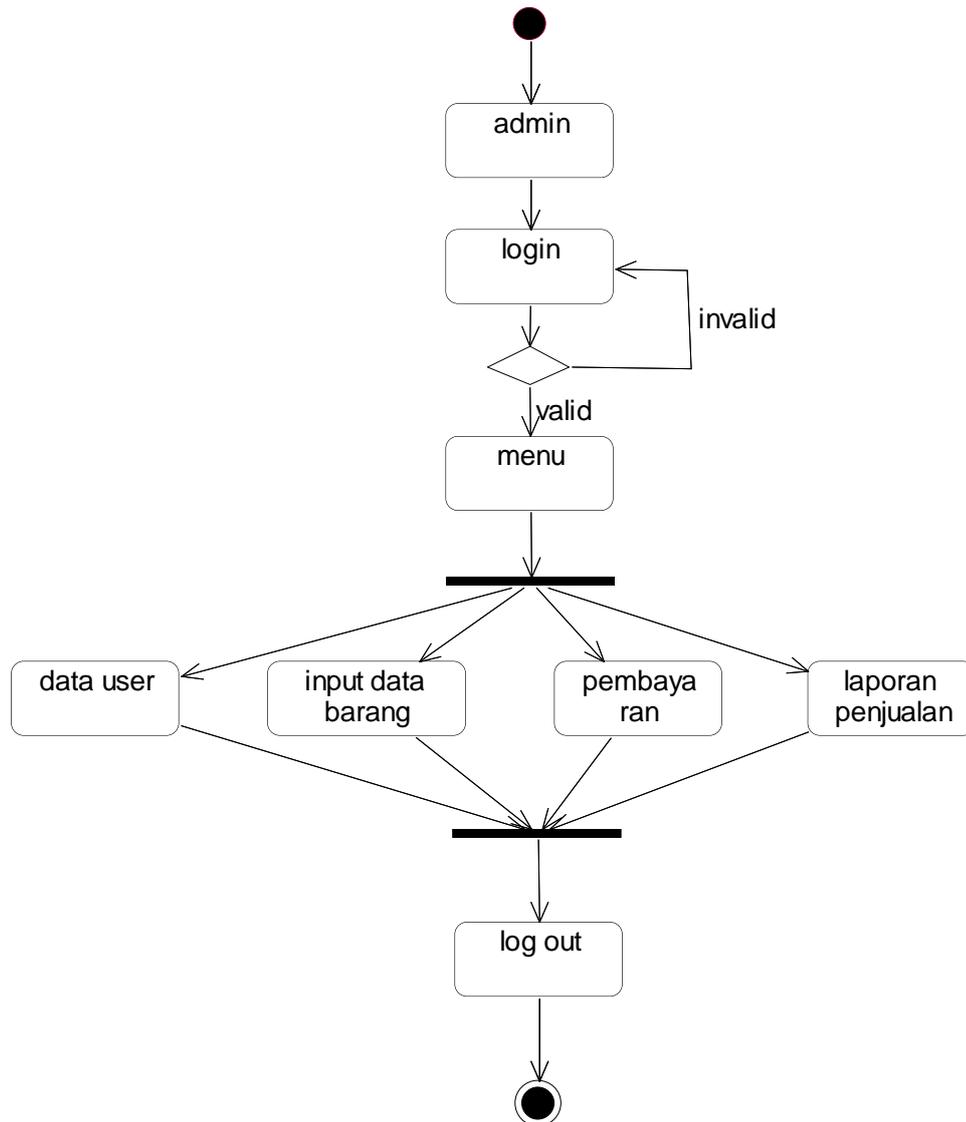


Gambar 3. 1 Use Case Diagram Admin

3. Activity Diagram

a. Activity Diagram Admin

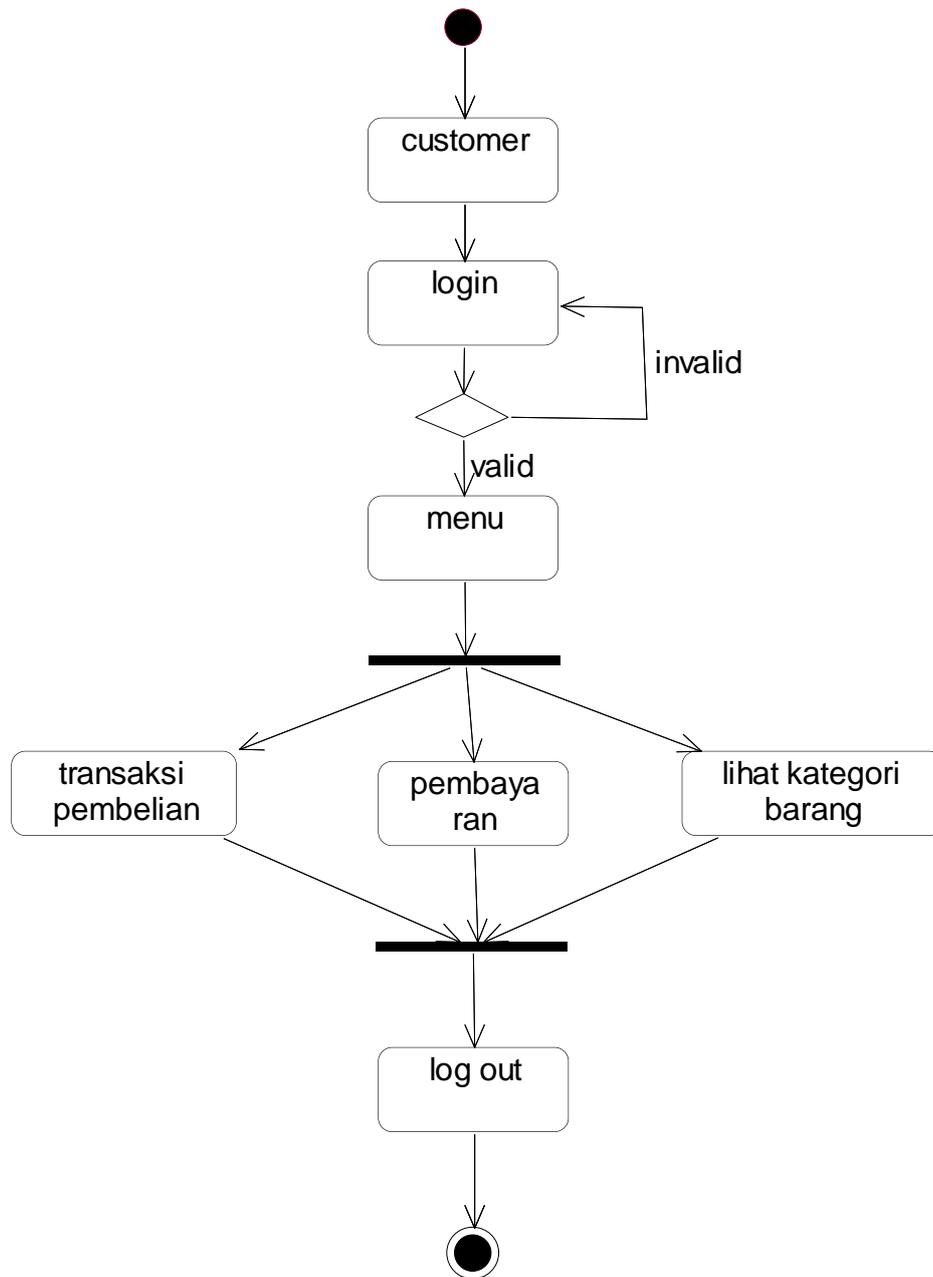
Activity Diagram pada admin ini dapat melakukan login terlebih dahulu, apabila telah melakukan login dengan benar maka akan masuk kedalam sistem dan dapat mengelola data user, mengimputkan data barang, kategori dan melihat laporan



Gambar 3. 2 Activity Diagram Admin

b. Activity Diagram Customer

Activity Diagram pada customer ini dapat melakukan login terlebih dahulu, apabila telah melakukan login dengan benar maka akan masuk kedalam sistem dan dapat ,melakukan transaksi dan melihat kategori barang.

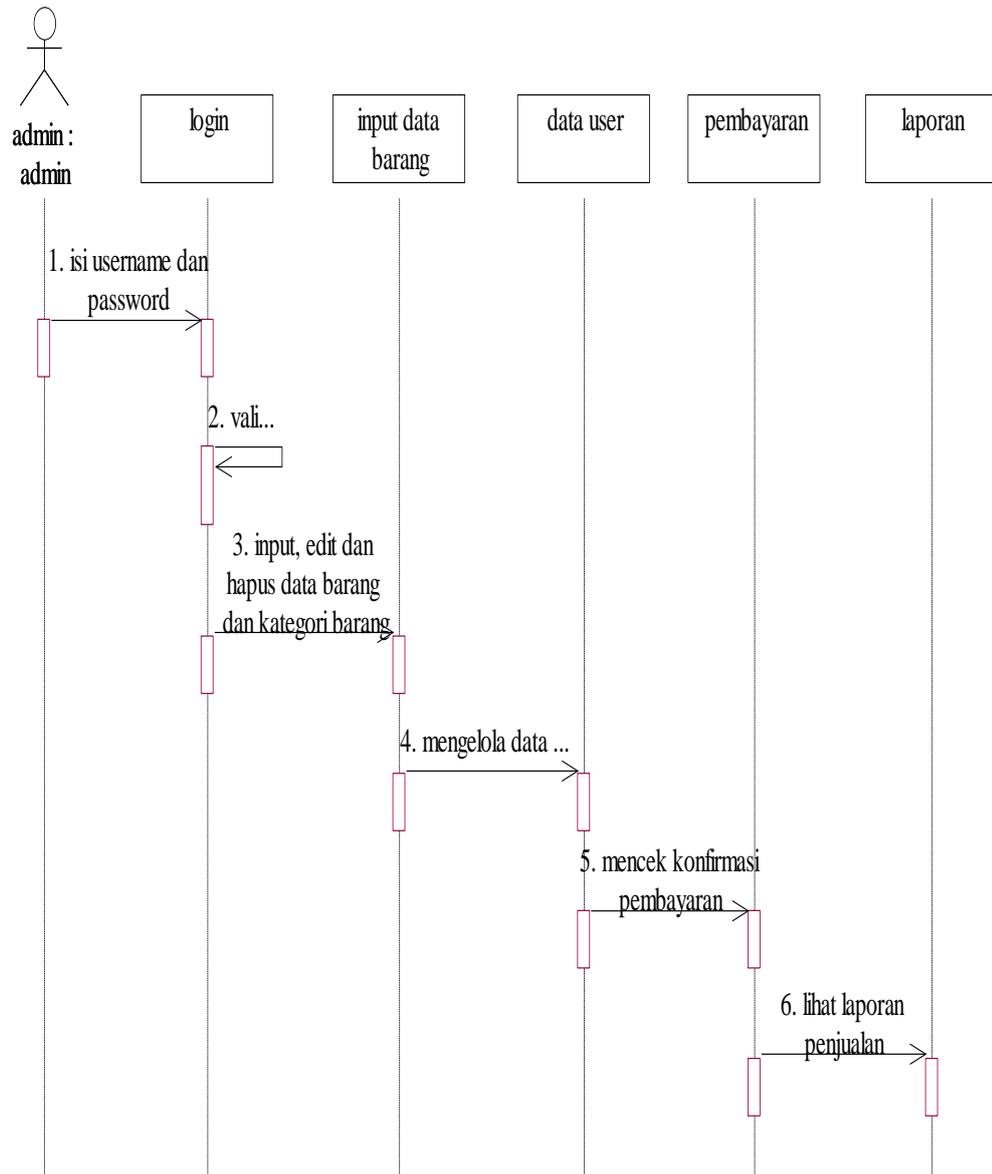


Gambar 3.3 Activity Diagram Customer

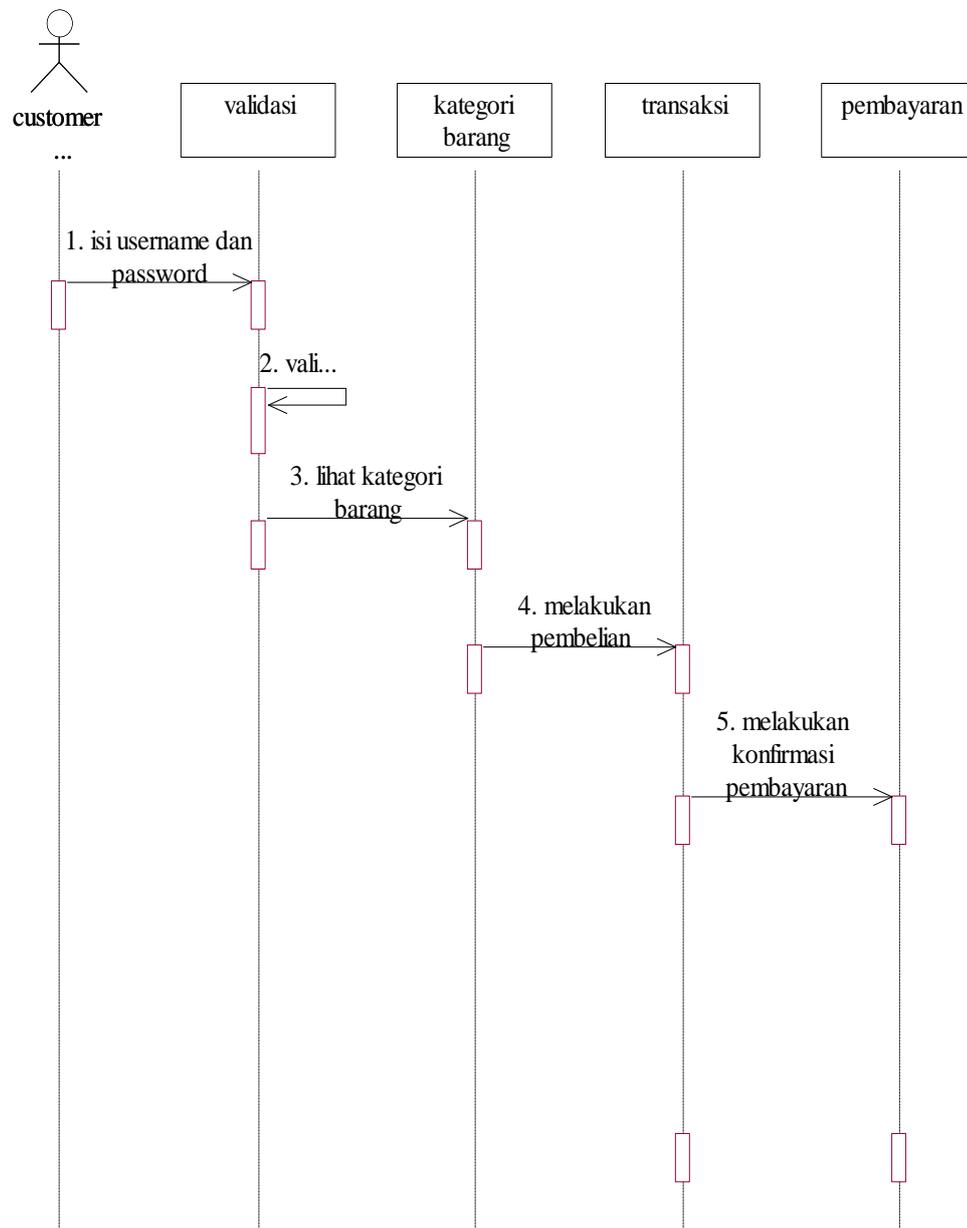
4. Sequence Diagram.

Sequence diagram pada menggambarkan interaksi antara objek yang ada disekitar sistem

a. *Sequence Diagram Admin*



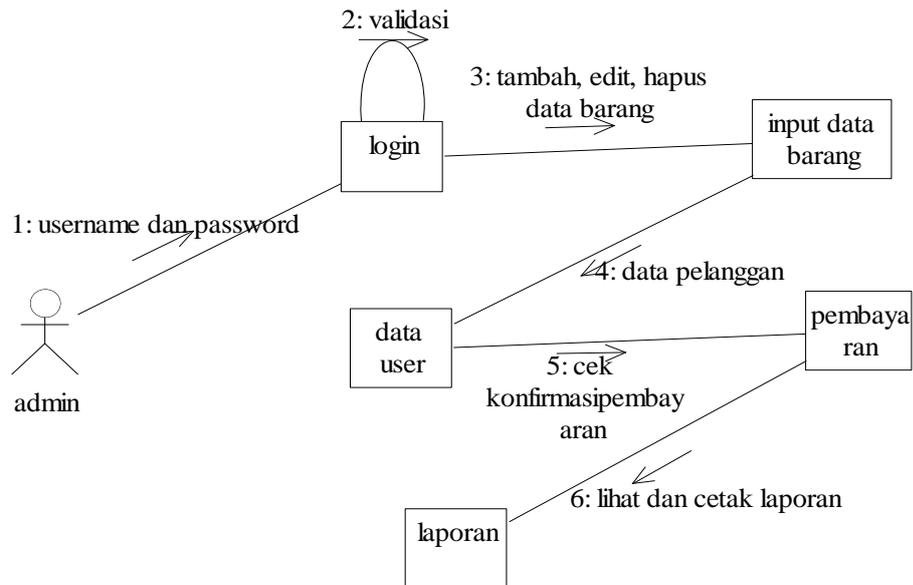
Gambar 3. 4 Sequence Diagram Admin

b. Sequence Diagram customer**Gambar 3. 5 Sequence Diagram Customer**

5. Collaboration diagram

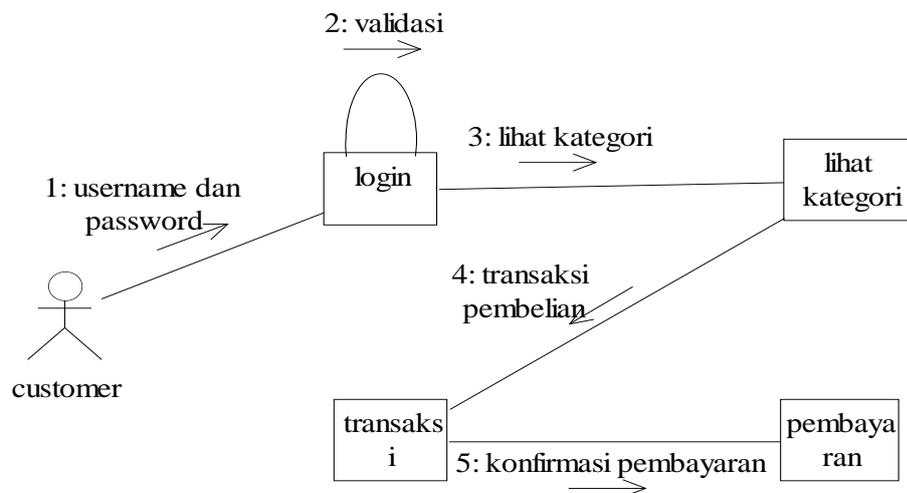
Collaboration diagram hampir sama dengan sequence diagram tetapi berbeda pada objek yang dititik tekankan, collaboration lebih menekankan pada pemunculan objek itu sendiri sedangkan sequence diagram lebih pada penyampaian message dengan parameter waktu.

a. Collaboration diagram admin



Gambar 3. 6 Collaboration Diagram Admin

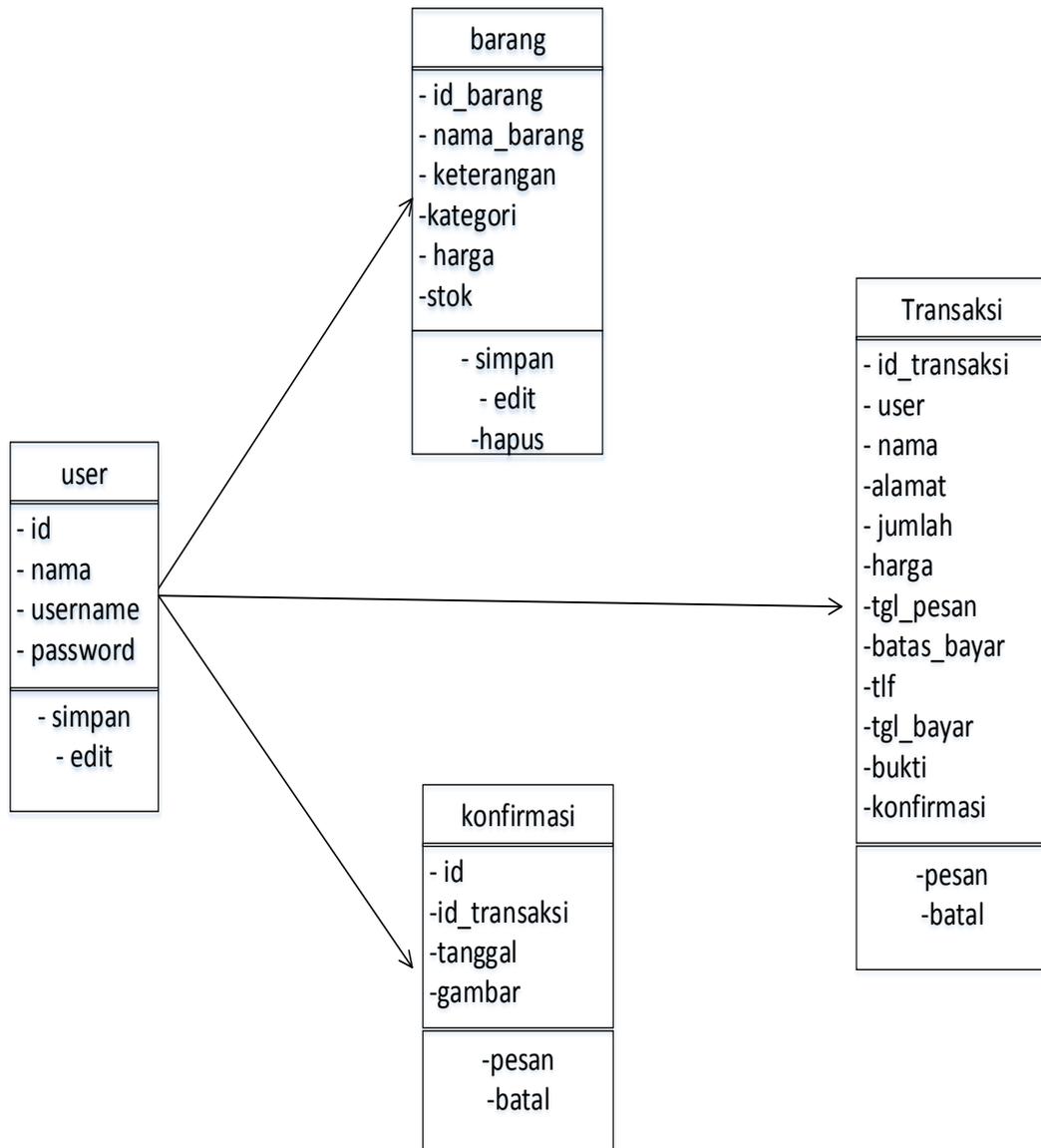
b. Collaboration diagram customer



Gambar 3. 7 Collaboration Diagram Customer

6. Class Diagram

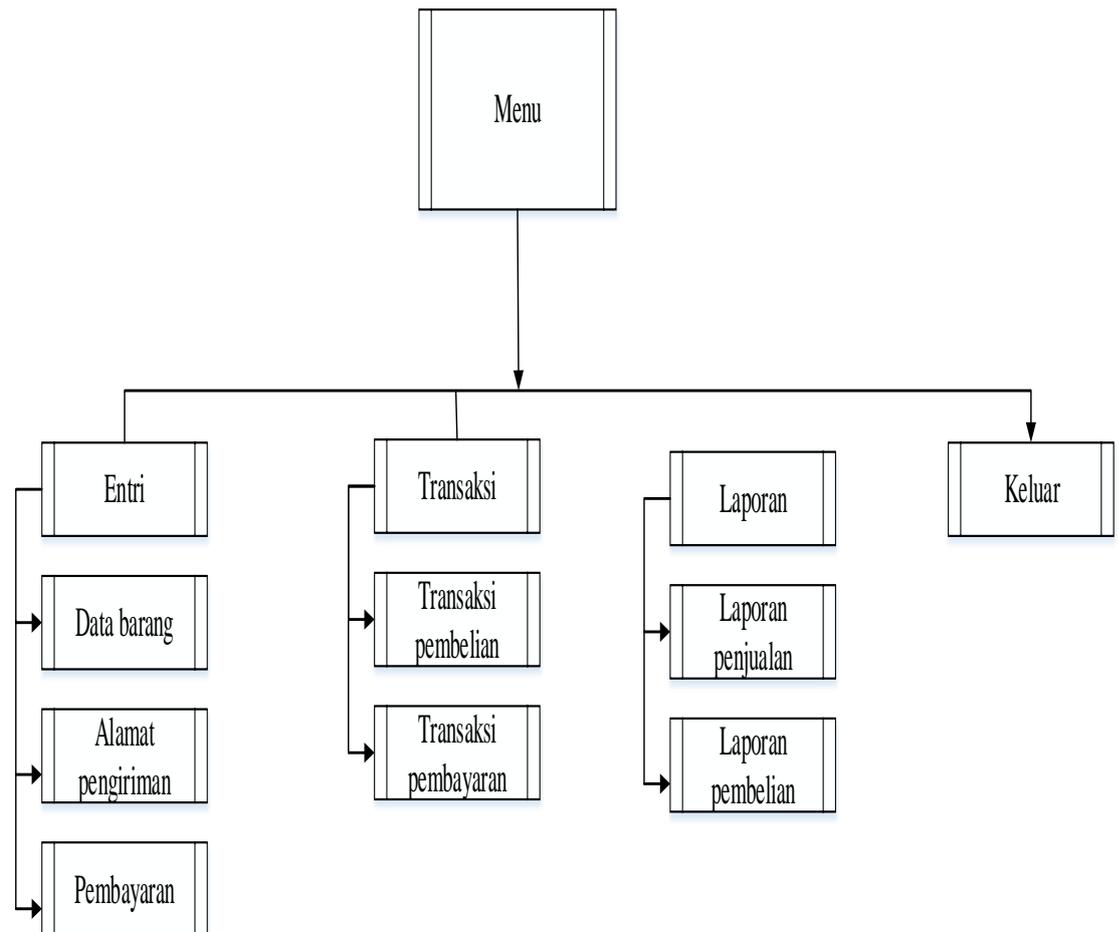
Class diagram menggambarkan struktur dari suatu sistem yang disajikan dalam bentuk class beserta atribut-atribut dan hubungan antar class. Umumnya class diagram dari suatu sistem akan menggambarkan juga bagaimana struktur database yang dibutuhkan untuk membangun sistem tersebut.



Gambar 3. 8 Class Diagram

7. Struktur Program

Desain struktur program merupakan suatu desain yang menggambarkan suatu hubungan modul program dengan modul program yang lainnya. Desain struktur program dari yang diusulkan oleh penulis dapat dilihat pada gambar berikut:



Gambar 3. 9 Struktur Program

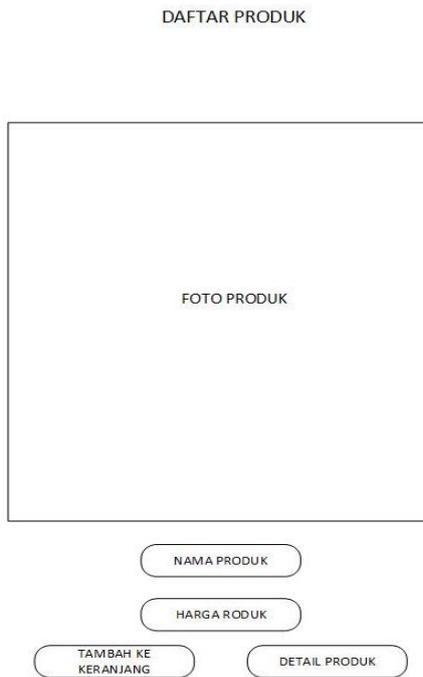
C. Desain Perancangan Terinci

Desain terperinci berfungsi untuk mengetahui lebih jelas tentang desain output, desain input, serta desain file.

1. Desain Output

Output (keluaran) adalah produk dari sistem informasi yang dapat terlihat. Output disini adalah tampilan di layar monitor. Adapun rancangan output pada aplikasi ini adalah sebagai berikut:

a. Daftar Produk



Gambar 3. 10 Daftar Produk

b. Data Barang

Data Barang					
No	Nama Barang	Keterangan	Kategori	Harga	Stok
X(11)	X(120)	X(250)	X(100)	X(11)	X(50)
/	/	/	/	/	/

Gambar 3. 11 Data Barang

c. **Keranjang Belanja**

Keranjang Belanja				
No	Nama Produk	Jumlah	Harga	Sub Total
X(11)	X(50)	X(3)	X(10)	X(11)

Gambar 3. 12 Keranjang Belanja

d. **Data Pemesanan Produk**

Pemesanan Produk				
No	Nama Pemesanan	Alamat	Tanggal Bayar	Bukti Bayar
X(11)	X(100)	X(100)		X(11)

Gambar 3. 13 Data Pemesanan Produk

e. Laporan Penjualan

Laporan Penjualan							
No	Id_barang	Nama	Alamat	Jumlah	Tanggal Bayar	Harga	Total Harga
X (11)	X (11)	X (120)	X (100)	X (128)	X (11)	X (11)	X (11)

Gambar 3. 14 Laporan Penjualan

2. Desain Input

Desain input berfungsi mengolah data menjadi informasi yang akan ditampilkan untuk pengunjung. Desain input pada perancangan sistem ini dapat terlihat pada gambar berikut ini:

a. Login

Login

Username

Password

Gambar 3. 15 Login

b. Tambah Data Barang

Tambah Data Barang	
Nama Barang	X (120)
Keterangan	X (250)
Kategori	X (100) V
Harga	X (50)
Stok	X (11)
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 3. 16 Tambah Data Barang

c. Alamat Pengiriman dan Pembayaran

Entri Alamat Pengiriman dan Pembayaran	
Pilih Provinsi	X (50) V
Pilih Kabupaten/Kota	X (50) V
Ekspedisi	X (50) V
Berat	X (50) gr
Pilih Bank	X (11) V
Alamat	X (120)
No Telepon	X (11)
<input type="button" value="Pesan"/> <input type="button" value="Batal"/>	

Gambar 3. 17 Alamat Pengiriman dan Pembayaran

3. Desain File

File merupakan kumpulan data-data atau record-record yang dibentuk oleh beberapa field. Data-data yang tersimpan dalam file ini seterusnya diproses oleh sistem pengolah data untuk menghasilkan output atau laporan yang nantinya dapat disajikan kepada pemakai laporan atau informasi yang dihasilkan tersebut.

Berdasarkan output yang telah dirancang serta bentuk input dari program yang nantinya akan diterapkan maka di disainlah file-file yang diperlukan untuk Sistem Informasi Penjualan ini yaitu:

a. Tabel User

Nama database : toko_online
 Nama tabel : user
 Primary key : id

Tabel 3. 2 User

Nama Field	Type	Lebar	Keterangan
Id	Integer	2	Id user
Nama	Varchar	50	Nama user
Username	Varchar	50	Username
Password	Varchar	128	Password user

b. Tabel Barang

Nama database : toko_online
 Nama tabel : barang
 Primary key : id_barang

Tabel 3. 3 Barang

Nama field	Type	Lebar	Keterangan
id_barang	Integer	11	Id barang
nama_barang	Varchar	120	Nama barang
Keterangan	Varchar	250	Keterangan
Kategori	Varchar	100	kategori
harga	Integer	11	Harga
Stok	Integer	50	Stok

gambar	Text		Gambar
size	Varchar	100	Size
warna	Varchar	100	Warna

c. Tabel Invoice

Nama database : toko_online

Nama tabel : invoice

Primary key : id_transaksi

Tabel 3. 4 Transaksi

Nama Field	Type	Lebar	Keterangan
Id_transaksi	Integer	11	Id invoice
Id_user	Integer	11	Id user
Nama	Varchar	100	Nama
Alamat	Varchar	100	Alamat
Jumlah	Varchar	128	Jumlah
Harga	Integer	128	Harga
Tgl_pesanan	Date		Tanggal pesanan
Batas_bayar	Date		Batas bayar
Tlf	Integer		No telephone
Tgl_bayar	Date	11	Tanggal bayar
Bukti	Varchar	128	Bukti
Konfirmasi	varchar	11	Konfirmasi

d. Tabel Konfirmasi

Nama database : toko_online

Nama tabel : konfirmasi

Primary key : id

Tabel 3. 5 Konfirmasi

Nama Field	Type	Lebar	Keterangan
Id	Integer	5	Id
Id_transaksi	Integer	20	Id invoice
Tanggal	Date	12	Tanggal

Gambar	Varchar	3	Gambar
--------	---------	---	--------

BAB IV

PENUTUP

A. Kesimpulan

Berdasarkan uraian dan penjelasan yang telah dikemukakan pada setiap bab dalam Laporan Tugas Akhir ini, maka dapat diambil beberapa kesimpulan yang diharapkan dapat bermanfaat bagi penulis, para pembaca dan khususnya pada Toko Sang Surya Serba Murah Batusangkar. Dari hasil penelitian dan kemudian dilanjutkan pada analisa dan disain sistem pengolahan data yang telah dilakukan, dapat ditarik kesimpulan antara lain:

1. Dengan penggunaan teknologi komputer secara optimal dapat membantu kinerja dari pihak Sang Surya Serba Murah Batusangkar dalam melakukan transaksi jual-beli, pemasaran produk dan pembuatan laporan penjualan yang lebih efektif dan efisien.
2. Dengan rancangan sistem yang baru, yang menggunakan aplikasi penjualan pada Sang Surya Serba Murah Batusangkar saat melakukan transaksi dan pembuatan laporan penjualan dapat membantu dalam meminimalkan kesalahan pengolahan data tersebut.
3. Dengan penerapan sistem baru ini semua data-data pengolahan data barang, data transaksi dan laporan penjualan pada Sang Surya Serba Murah Batusangkar akan tersimpan dalam sebuah file database, sehingga akan memudahkan dalam proses pencarian data.

Dalam melakukan suatu penelitian terhadap suatu sistem baik untuk mengembangkan maupun membuat sistem baru maka hal-hal yang perlu diperhatikan adalah bagaimana prosedur pelaksanaan dari pengembangan sistem tersebut, agar hasil dari analisa dapat diterapkan dengan baik. Adapun hal yang sangat menentukan dari pelaksanaan sistem adalah orang-orang yang menjalankan sistem tersebut, sebab sebaik apapun sistem yang dibuat jika tidak didukung pemakai dengan baik tidak akan mendapatkan hasil yang diinginkan.

B. Saran

Untuk tercapainya efektifitas dan efisiensi kerja dengan ditunjang sistem pengolahan data yang baik sesuai dengan tujuan Dapat meningkatkan mutu kinerjanya, maka berikut ini penulis menganjurkan beberapa saran-saran antara lain :

1. Aplikasi penjualan ini sebaiknya digunakan toko agar bias meningkatkan pendapatan.
2. Sebaiknya ketika sebelum menggunakan aplikasi ini diiklankan dulu leawat sosial media atau yang lainnya, agar bias menjangkau lebih banyak pelanggan.
3. Melakukan penyesuaian sistem baru terhadap sistem lama.
4. Melakukan pelatihan terhadap tenaga kerja yang akan menjalankan sistem baru.
5. Dengan sistem penyajian informasi yang baru, pemakai disarankan untuk memperhatikan kekurangan sistem agar segera dicari pemecahan masalah dan melakukan perbaikan atau pengembangan untuk mencapai kesempurnaan.

DAFTAR PUSTAKA

- Anonymous. Aplikasi Manajemen Database Pendidikan Berbasis Web dengan PHP dan MySQL. Yogyakarta: Andi, 2005.
- Arifin Hidayat, Rizal. 2012. “ Web E-commerce pada tripio komputer menggunakan pendekatan busines to customer”.(2 Agustus):60-70
- Ginting,Elizaandayni.2013.Aplikasi Penjualan Berbasis Web (E-commerce) Menggunakan Joomla Pada Mutiara Fashion.Bandung:Universitas Widyatama.
- Hakim, Lukmanul dan Uus Musalini. 2004. Cara Cerdas Menguasai Layout, Desain dan Aplikasi Web . (Jakarta: PT Elex Media Komputindo)
- Hasan,noor. 2014.”Perancangan Aplikasi E-commerce Penjualan Komputer Pada Toko Mitra Purworejo Menggunakan PHP dan Mysql”(1 Maret):81-87.
- Jogiyanto. Analisis dan Desain Sistem Informasi Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Yogyakarta: CV. Andi Offset, 2005.
- Madcoms. Dreamweaver CS5 PHP-MySQL, Yogyakarta : Andi, 2012
- Mustofa, Ahmad and Mutmainah. 2015.”Perancangan e commerce penjualan komputer dan alat elektronik berbasis web pada toko damar komputer pringsewu” (4 juli):62-67.
- Nugroho, A. Analisis Dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek. Bandung: Informatika, 2005.
- Rachman,raditya.2014.”Pengertian Program Paket(Perangkat Lunak Aplikasi)”
- Wahyono, T. Sistem Informasi: Konsep Dasar, Analisis dan Desain dan Implementasi. Yogyakarta: Graha Ilmu, 2004.

LAMPIRAN

KEMENTERIAN PERTANIAN DAN PERIKANAN
REPUBLIK INDONESIA

KEHATI-HATI TERANGAN TENTANG BERAKSIKAN PERMITIFAS

Kemampuan beraktilitas Permis Perikanan (KBP) adalah izin yang diberikan kepada nelayan yang telah memenuhi persyaratan teknis dan administratif yang ditetapkan dalam peraturan perundang-undangan.

Jenis: PERMITS PERIKANAN

Nomor: 1000/00000

Jenis Permis: 1. Untuk Berperikanan dalam Wilayah Perairan Nelayan
2. Untuk Berperikanan dalam Wilayah Perairan Nelayan

Untuk persyaratan teknis dan administratif yang ditetapkan dalam peraturan perundang-undangan.

Permis ini hanya berlaku untuk wilayah perikanan yang ditetapkan dalam peraturan perundang-undangan.

Harmonis, S.P., M.A., M.P.
Deputi Kepala Negara
Perikanan

Jakarta, 10/10/2010



KEMENTERIAN AGAMA
INSTITUT AGAMA ISLAM NEGERI RADJASA NGABEAN
Jl. Sekeloa Tengah No. 101, Radjasa Ngabean, Kabupaten Sukoharjo, Jawa Tengah 50132
Telp. (0271) 8511000

KALAH BONGSI HIRING
KEGIATAN RUBINGAN PENULISAN TUGAS AKHIR

Tema: Kajian Sistematis
Waktu: 1000-1001
Tempat: Ruang Pertemuan
Waktu: Pukul 08.00 s.d. 10.00
Instansi: Majelis Ulama Indonesia (MUI) Kabupaten Sukoharjo
Instansi: Majelis Ulama Indonesia (MUI) Kabupaten Sukoharjo
Instansi: Majelis Ulama Indonesia (MUI) Kabupaten Sukoharjo

No	NAMA PESERTA	PUNDIUMEN	Poin
1	Johan ^{21/05/2001}	Kelembagaan, Struktur, dan Fungsi	100
2	Johan ^{21/05/2001}	KAN & (Lingkungan Tersebut)	100
3	Johan ^{21/05/2001}	Kelembagaan & Fungsional Organisasi	100
4	Johan ^{21/05/2001}	Solusi, Analisis & Model	100
5	Johan ^{21/05/2001}	Struktur Organisasi, Input, Output, dan Proses	100
6	Johan ^{21/05/2001}	Peran, Fungsi dan Struktur Organisasi	100
7	Johan ^{21/05/2001}	Kelembagaan, Model & Analisis & Repair	100
8	Johan ^{21/05/2001}	Struktur, Fungsi dan Struktur Organisasi	100

Disusun oleh: Johan^{21/05/2001}
 Tanggal: 10/05/2023
 Di: Sukoharjo


Ketua Panitia
Johan^{21/05/2001}
 NIP. 123456789


Wakil Ketua Panitia
Johan^{21/05/2001}
 NIP. 123456789


Sekretaris Panitia
Johan^{21/05/2001}
 NIP. 123456789

LISTING PROGRAM

```
<?php
```

```
defined('BASEPATH') OR exit('No direct script access allowed');
```

```
class Dashboard extends CI_Controller {
```

```
    public function __construct(){
```

```
        parent::__construct();
```

```
        $this->load->library('form_validation');
```

```
        if ($this->session->userdata('role_id')!= '2') {
```

```
            $this->session->set_flashdata('pesan','<div class="alert alert-  
danger alert-dismissible fade show"role="alert">
```

```
                Anda Belum Login
```

```
        }
```

```
    public function tambah_ke_keranjang($id){
```

```
        $barang = $this->model_barang->find($id);
```

```
        $jumlah = $this->input->post('jumlah');
```

```
        $data = array(
```

```
            'id'      => $barang->id_barang,
```

```
            'warna'   =>   $barang->warna,
```

```
            'size'    =>   $barang->size,
```

```
            'qty'     =>   $jumlah,
```

```
            'price'   =>   $barang->harga,
```

```
    'name' => $barang->nama_barang,  
);
```

```
$this->cart->insert($data);  
redirect('home');  
}
```

```
public function detail_keranjang(){  
    $this->load->view('template/header');  
    $this->load->view('template/sidebar');  
    $this->load->view('keranjang');  
    $this->load->view('template/footer');  
}
```

```
public function hapus_keranjang(){  
    $this->cart->destroy();  
    redirect('home');  
}
```

```
public function pembayaran(){  
    $data['ongkir'] = "  
    if(count($_POST)){
```

```
    tps://api.rajaongkir.com/starter/cost",
```

```
        CURLOPT_RETURNTRANSFER => true,
```

```
        CURLOPT_ENCODING => "",
        CURLOPT_MAXREDIRS => 10,
        CURLOPT_TIMEOUT => 30,
        CURLOPT_HTTP_VERSION =>
CURL_HTTP_VERSION_1_1,
        CURLOPT_CUSTOMREQUEST => "POST",
        CURLOPT_POSTFIELDS => "origin=453&destination=" . $this->input->post('kota'). "&weight=" . $this->input->post('berat'). "&courier=" . $this->input->post('ekspedisi'),
        CURLOPT_HTTPHEADER => array(
            "content-type: application/x-www-form-urlencoded",
            "key: 0b3bb757b6503a84a5d1856567f76855"
        ),
    }
}
```

```
$data['user'] = $this->db->get_where('tb_user', ['username' => $this->session->userdata('username')])->row_array();
```

```
$data['barang'] = $this->db->get('tb_barang')->row_array();
```

```
$this->load->view('template/header');
```

```
$this->load->view('template/sidebar');
```

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

```
$this->load->view('template/footer');
```

```
}
```

```

public function kota($provinsi){

    $curl = curl_init();

    curl_setopt_array($curl, array(

        CURLOPT_URL =>
"https://api.rajaongkir.com/starter/city?&province=".$provinsi,

        CURLOPT_RETURNTRANSFER => true,

        CURLOPT_ENCODING => "",

        CURLOPT_MAXREDIRS => 10,

        CURLOPT_TIMEOUT => 30,

        CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,

        CURLOPT_CUSTOMREQUEST => "GET",

        CURLOPT_

echo "cURL Error #:" . $err;

    } else {

        $kota = json_decode($response,true);

        if($kota['rajaongkir']['status']['code'] == '200'){

            echo "<option value=">Pilih Kota</option>";

            foreach($kota['rajaongkir']['results'] as $kt){

                echo "<option
value='$kt[city_id]'">$kt[city_name]</option>";

            }

        }

    }

}

```

```

public function pesanan()
{
    // $id = $this->session->userdata('id')->row_array();

    // $data['pesanan'] = $this->db->get_where('tb_invoice',['id'] = $id)-
    >row_array();

    $this->load->view('template/header');

    $this->load->view('template/sidebar');

    $this->load->view('proses_pesanan');

    $this->load->view('template/footer');
}

{
    $this->load->view('template/header');

    $this->load->view('template/sidebar');

    $this->load->view('konfirmasi');

    $this->load->view('template/footer');
}

public function tambah_konfirmasi()
{
    $this->load->library('upload');

    $data['user'] = $this->db->get_where('tb_invoice', ['nama' => $this-
    >session->userdata('username')])->row_array();

    $this->form_validation->set_rules('id','Id','trim|required');

```

```
if($this->form_validation
ets/buktiBayar/';

$this->upload->initialize($config);

if ($this->upload->do_upload('bukti')) {
    $upload_data = $this->upload->data('file_name');
    $this->db->set('bukti', $upload_data);
} else {
    echo $this->upload->display_errors();
}
}

// var_dump($upload_file);

// die();

$this->db->set('tgl_bayar',$tgl_bayar);
$this->db->set('konfirmasi', $konfirmasi);
$this->db->where('id',$id);
$this->db->update('tb_invoice');

$this->session->set_flashdata('pesan','Berhasil Melakukan
Pembayaran');
```

```
date_default_timezone_set('Asia/Jakarta');

$id_user          = $this->input->post('id_user');
$name            = $this->input->post('nama');
$alamat         = $this->input->post('alamat');
$jumlah        = $this->input->post('jumlah');
$harga         = $this->input->post('harga');
$tlp           = $this->input->post('tlp');
$ongkos        = $this->input->post('ongkos');
```

```
$this->form_
```

```
    'tgl_pesan' => date('Y-m-d H:i:s'),
    'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),
date('s'),date('m'),date('d') + 1,date('Y'))),
    'tlp' => $tlp,
    'tgl_bayar' =>0,
    'bukti' => '-',
);
```

```
$id          = $this->input->post('id_barang');
```

```
$stock = $this->input->post('stok');
```

```
$kurangStock = $stock - $jumlah;
```

```
$this->db->set('stok',$kurangStock);
```

```
$this->db->where('id_barang',$id);
```

```
$this->db->update('tb_barang');

$this->db->insert('tb_invoice',$invoice);

$this->cart->destroy();

$this->session->set_flashdata('pesan','Pesanan Sudah Dikirim');

redirect(
    </div>');
    redirect('auth/login');
}
}

public function index()
{
    $data['user'] = $this->model_user->tampil_data()->result();
    $this->load->view('template_admin/header');
    $this->load->view('template_admin/sidebar');
    $this->load->view('admin/user',$data);
    $this->load->view('template_admin/footer');
}

public function ganti_password($id)
{
    $data['user'] = $this->model_user->tampil_data()->result();
    $data['id'] = $this->db->get_where('tb_user',['id' =>$id])->row_array();
    $this->load->view('template_admin/header');
    $this->load->view('template_admin/sidebar');
```

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

$this->load->view('template_admin/footer');

}

public function changePassword()
{
    // 'id' => $this->input->post('id'),
    // ];

    $this->db->set('password', $password);

    $this->db->where('id', $id);

    $this->db->update('tb_user');

    $this->session->unset_userdata('username');

    $this->session->set_flashdata('message', '<div class="alert alert-success"
    role="alert">Password has been change! Please login.</div>');

    redirect('admin/user');
}
}
}
```

<?php

```

class Registrasi extends CI_Controller{

    public function index()
    {
        $this->form_validation->set_rules('nama','Nama','required',[
            'required'=> 'Nama Wajib di isi'

        ]);

        $this->form_validation->set_rules('username','Username','required',[
            'required'=> 'Username wajib di isi'

        ]);

        'id'          =>    "",
        'nama'         =>    $this->input->post('nama'),
        'username'     =>    $this->input->post('username'),
        'password'     =>    password_hash($this->input-
        >post('password_1'), PASSWORD_DEFAULT),
        'role_id'      =>    2,

    );

    $this->db->insert('tb_user',$data);

    redirect('auth/login');

    }

    }

}

```

```

<?php
public function index()
{
    $data['barang'] = $this->model_review->tampil_data()->result();
    $this->load->view('template_admin/header');
    $this->load->view('template_admin/sidebar');
    $this->load->view('detail_barang',$data);
    $this->load->view('template_admin/footer');
}
public function tambah_aksi()
{
ar
    );
    $this->model_barang->tambah_barang($data,'tb_barang');
    redirect('admin/data_barang/index');
}

<?php
class Dashboard_admin extends CI_Controller{

    public function __construct(){

```

```
parent::__construct();

if ($this->session->userdata('role_id')!= '1') {

    $

barang);

    $this->load->view('template_admin/header');

    $this->load->view('template_admin/sidebar');

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

    $this->load->view('template_admin/footer');

}

}

?>
```

```
<?php

class Model_barang extends CI_Model{

    public function tampil_data(){

        return $this->db->get('tb_barang');

    }

    public function tambah_barang($data,$table){

        $this->db->insert($table,$data);

    }

}
```

```

public function hapus($where,$table){
    $this->db->where($where);
    $this->db->delete($table);
}

public function edit_barang($where,$table)
{
    return $this->db->get_where($table,$where);
}

public function update_data($where,$data,$table)
{
    $this->db->where($where);

}

public function detail_barang($id_barang){
    $result= $this->db->where('id_barang',$id_barang)->get('tb_barang');
    if($result->num_rows()>0){
        return $result->result();
    }else{
        return false;
    }
}
}

```

?>

```
<?php
```

```
class Model_invoice extends CI_Model{
```

```
        'id_barang'                => $item['id'],
```

```
        'nama_barang'             => $item['name'],
```

```
        'jumlah'                  => $item['qty'],
```

```
        'harga'                   => $item['price'],
```

```
    );
```

```
    $this->db->insert('tb_pesanan', $data);
```

```
    }
```

```
    return TRUE;
```

```
}
```

```
public function tampil_data()
```

```
{
```

```
    $result = $this->db->get_where('tb_invoice',['bukti' => 1]);
```

```
    if($result->num_rows() > 0){
```

```
        return $result->result();
```

```
    }else{
```

```
        return false;
```

```
    }
```

```
}
```

```
public function ambil_id_invoice($id_invoice)
```

```
{
```

```

$result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');
if($result->num_rows() > 0){
    return $result->row();

}

}else{
    return false;
}
}

public function ambil_id_pesanan($id_invoice)
{
?php
class Model_invoice extends CI_Model{
    public function index(){
        date_default_timezone_set('Asia/Jakarta');
        $nama      = $this->input->post('nama');
        $alamat    = $this->input->post('alamat');

        $invoice = array (
            'nama' => $nama,
            'alamat' => $alamat,
            'tgl_pesan' => date('Y-m-d H:i:s'),
            'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),
date('s'),date('m'),date('d') + 1,date('Y'))),

        );

```

```

$this->db->insert('tb_invoice',$invoice);

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

foreach ($this->cart->contents() as $item){

    $data = array(

        'id_invoice'          => $хid_invoice,

        'id_barang'           => $item['id'],

        'nama_barang'         => $item['name'],

        'jumlah'               => $item['qty'],

        'harga'                => $item['price'],

    );

    $this->db->insert('tb_pesanan', $data);

}

return TRUE;

}

public function tampil_data()

{

    $result = $this->db->get_where('tb_invoice',['bukti' => 1]);

    if($result->num_rows() > 0){

        return $result->result();

    }else{

        return false;

    }

}

}

public function ambil_id_invoice($хid_invoice)

```

```
{
    $result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');
    if($result->num_rows() > 0){
        return $result->row();

    }else{
        return false;
    }
}

public function ambil_id_pesanan($id_invoice)
{
    $result = $this->db->where('id_invoice',$id_invoice)->get('tb_pesanan');
    if($result->num_rows() > 0){
        return $result->result();

    }else{
        return false;
    }
}
}

?>
```

```

<?php
class Model_invoice extends CI_Model{
    public function index(){
        date_default_timezone_set('Asia/Jakarta');
        $nama      = $this->input->post('nama');
        $alamat    = $this->input->post('alamat');

        $invoice = array (
            'nama' => $nama,
            'alamat' => $alamat,
            'tgl_pesan' => date('Y-m-d H:i:s'),
            'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),
date('s'),date('m'),date('d') + 1,date('Y'))),
        );
        $this->db->insert('tb_invoice',$invoice);
        $id_invoice = $this->db->insert_id();

        foreach ($this->cart->contents() as $item){
            $data = array(
                'id_invoice'      => $id_invoice,
                'id_barang'      => $item['id'],
                'nama_barang'    => $item['name'],
                'jumlah'         => $item['qty'],
                'harga'          => $item['price'],
            );
        }
    }
}

```

```

        );
        $this->db->insert('tb_pesanan', $data);
    }
    return TRUE;
}

public function tampil_data()
{
    $result = $this->db->get_where('tb_invoice',['bukti' => 1]);
    if($result->num_rows() > 0){
        return $result->result();
    }else{
        return false;
    }
}

public function ambil_id_invoice($id_invoice)
{
    $result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');
    if($result->num_rows() > 0){
        return $result->row();
    }else{
        return false;
    }
}

public function ambil_id_pesanan($id_invoice)

```

```

{
    $result = $this->db->where('id_invoice',$id_invoice)->get('tb_pesanan');
    if($result->num_rows() > 0){
        return $result->result();

    }else{
        return false;
    }
}
}

```

?>

?php

```

class Model_invoice extends CI_Model{
    public function index(){
        date_default_timezone_set('Asia/Jakarta');
        $nama      = $this->input->post('nama');
        $alamat    = $this->input->post('alamat');

        $invoice = array (
            'nama' => $nama,
            'alamat' => $alamat,
            'tgl_pesan' => date('Y-m-d H:i:s'),
            'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),
            date('s'),date('m'),date('d') + 1,date('Y'))),

```

```

);

$this->db->insert('tb_invoice',$invoice);

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

foreach ($this->cart->contents() as $item){

    $data = array(

        'id_invoice'          => $id_invoice,

        'id_barang'          => $item['id'],

        'nama_barang'        => $item['name'],

        'jumlah'              => $item['qty'],

        'harga'               => $item['price'],

    );

    $this->db->insert('tb_pesanan', $data);

}

return TRUE;

}

public function tampil_data()

{

    $result = $this->db->get_where('tb_invoice',['bukti' => 1]);

    if($result->num_rows() > 0){

        return $result->result();

    }else{

        return false;

    }

}

```

```
}  
public function ambil_id_invoice($id_invoice)  
{  
    $result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');  
    if($result->num_rows() > 0){  
        return $result->row();  
  
    }else{  
        return false;  
    }  
}  
public function ambil_id_pesanan($id_invoice)  
{  
    $result = $this->db->where('id_invoice',$id_invoice)->get('tb_pesanan');  
    if($result->num_rows() > 0){  
        return $result->result();  
  
    }else{  
        return false;  
    }  
}  
}  
  
?>
```

```
<?php
```

```
class Model_invoice extends CI_Model{
```

```
    public function index(){
```

```
        date_default_timezone_set('Asia/Jakarta');
```

```
        $nama      = $this->input->post('nama');
```

```
        $alamat    = $this->input->post('alamat');
```

```
        $invoice = array (
```

```
            'nama' => $nama,
```

```
            'alamat' => $alamat,
```

```
            'tgl_pesan' => date('Y-m-d H:i:s'),
```

```
            'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),  
date('s'),date('m'),date('d') + 1,date('Y'))),
```

```
        );
```

```
        $this->db->insert('tb_invoice',$invoice);
```

```
        $id_invoice = $this->db->insert_id();
```

```
        foreach ($this->cart->contents() as $item){
```

```
            $data = array(
```

```
                'id_invoice'      => $id_invoice,
```

```
                'id_barang'       => $item['id'],
```

```
                'nama_barang'     => $item['name'],
```

```

                'jumlah'                => $item['qty'],
                'harga'                 => $item['price'],
            );
            $this->db->insert('tb_pesanan', $data);
        }
        return TRUE;
    }
    public function tampil_data()
    {
        $result = $this->db->get_where('tb_invoice',['bukti' => 1]);
        if($result->num_rows() > 0){
            return $result->result();
        }else{
            return false;
        }
    }
    public function ambil_id_invoice($id_invoice)
    {
        $result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');
        if($result->num_rows() > 0){
            return $result->row();
        }else{
            return false;
        }
    }

```

```

    }
    public function ambil_id_pesanan($id_invoice)
    {
        $result = $this->db->where('id_invoice',$id_invoice)->get('tb_pesanan');
        if($result->num_rows() > 0){
            return $result->result();

        }else{
            return false;
        }
    }
}

?>

```

<?php

```

class Model_invoice extends CI_Model{
    public function index(){
        date_default_timezone_set('Asia/Jakarta');

        $nama      = $this->input->post('nama');
        $alamat    = $this->input->post('alamat');

        $invoice = array (

```

```

        'nama' => $nama,

        'alamat' => $alamat,

        'tgl_pesan' => date('Y-m-d H:i:s'),

        'batas_bayar' => date('Y-m-d H:i:s', mktime(date('H'),date('i'),
date('s'),date('m'),date('d') + 1,date('Y'))),

    );

    $this->db->insert('tb_invoice',$invoice);

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

    foreach ($this->cart->contents() as $item){

        $data = array(

            'id_invoice'          => $id_invoice,

            'id_barang'           => $item['id'],

            'nama_barang'         => $item['name'],

            'jumlah'               => $item['qty'],

            'harga'                => $item['price'],

        );

        $this->db->insert('tb_pesanan', $data);

    }

    return TRUE;

}

public function tampil_data()

{

    $result = $this->db->get_where('tb_invoice',['bukti' => 1]);

```

```

        if($result->num_rows() > 0){
            return $result->result();
        }else{
            return false;
        }
    }
}

public function ambil_id_invoice($id_invoice)
{
    $result = $this->db->where('id',$id_invoice)->limit(1)->get('tb_invoice');
    if($result->num_rows() > 0){
        return $result->row();
    }else{
        return false;
    }
}

public function ambil_id_pesanan($id_invoice)
{
    $result = $this->db->where('id_invoice',$id_invoice)->get('tb_pesanan');
    if($result->num_rows() > 0){
        return $result->result();
    }else{
        return false;
    }
}

```

```
}  
}  
  
?>
```

```
<div class="container-fluid">
```

```
<center><p><h3>LAPORAN PENJUALAN</h3></p>
```

```
<P><h3>TOKO SANG SURYA SERBA MURAH</h3></p>
```

```
<table class="table table-bordered table-striped table-hover " border="1">
```

```
<tr>
```

```
<th>ID BARANG</th>
```

```
<th>NAMA</th>
```

```
<th>ALAMAT</th>
```

```
<th>JUMLAH</th>
```

```
<th>TANGGAL_BAYAR</th>
```

```
<th>HARGA</th>
```

```
<th>TOTAL HARGA</th>
```

```
</tr>
```

```
<?php
```

```
$total = 0;
```

```
foreach ($laporan as $lp) :
```

```
$subtotal = $lp->jumlah * $lp->harga;
```

```
$total += $subtotal;
```

```
?>
```

```
<tr align="center">
```

```
<td><?= $lp->id?></td>
```

```
<td><?= number_format($subtotal,0,',','.')?></td>
```

```
</tr>
```

```
<?php endforeach; ?>
```

```
<tr>
```

```
<td colspan="6" align="right">Grand Total </td>
```

```
<td align="right"> Rp. <?= number_format($total,0,',','.') ?></td>
```

```
</tr>
```

```
</table>
```

```
</center>
```

```
</div>
```

```
<script >
```

```
    window.print();
```

```
</script>
```

```

<div class="container-fluid">
  <!-- Content Row -->
  <p><h2>Selamat Datang Admin!</h2></p>
  <div class="row">

    <!-- Earnings (Monthly) Card Example -->
    <div class="col-xl-3 col-md-6 mb-4">
      <div div>
        <div class="col-auto">
          <i class="fas fa-calendar fa-2x text-gray-300"></i>
        </div>
      </div>
    </div>
  </div>
</div>

  <!-- Earnings (Monthly) Card Example -->
  <div class="col-xl-3 col-md-6 mb-4">
    <div class="card border-left-success shadow h-100 py-2">
      <div class="card-body">
        <div class="row no-gutters align-items-center">
          <div class="col mr-2">
            <div class="text-xs font-weight-bold text-success text-
uppercase mb-1">

```