

# **RANCANG BANGUN APLIKASI *E-COMMERCE* PADA CV. SAMMARINDO BERBASIS WEB**

Oleh :

Tiur Sarina Sianipar. [c.tiur@yahoo.co.id](mailto:c.tiur@yahoo.co.id)

Pembimbing I : Dr. Marzuki Silalahi, M.T

Pembimbing II : Yulhendri, S.T, M.T

## **ABSTRAK**

CV. SAMMARINDO merupakan perusahaan yang bergerak dalam bidang industri pembuatan sepatu kulit. Metode transaksi penjualan saat ini masih manual sehingga masih terbatas dengan waktu dan tempat. Selain itu, pemasaran produknya dilakukan secara konvensional yaitu via telepon/sms, *face to face* serta menggunakan media sosial. Cara ini kurang efektif untuk menjangkau pangsa pasar yang luas.

Metode yang digunakan dalam perancangan aplikasi *E-Commerce* ini adalah menggunakan metode UML sedangkan untuk pengembangan sistemnya menggunakan metode *eXtreme Programming* yang terdiri dari *planning, design, coding, testing*. Hasil yang ingin dicapai adalah *e-commerce* berbasis WEB, dimana sistem ini dapat memudahkan konsumen untuk mendapatkan informasi yang diinginkan secara lengkap, melakukan transaksi dengan mudah dan cepat tanpa dibatasi ruang dan waktu. Serta membantu perusahaan menjangkau pasar yang lebih luas dengan adanya sistem penjualan secara *online*.

Kata kunci : *E-commerce*, PHP, MySQL

## **ABSTRACT**

*CV SAMMARINDO is a company that budges on the industry field especially in the production of leather shoes. Today, the method of selling transaction still use conventional method in which it is still limited with time and place. Moreover, the product marketing was held in conventional way through telephone, message, face-to-face, and using social media. However, this way is still ineffective to extend the broader market. Method that used in designing the E-commerce application is using the UML Method. Meanwhile, to develop the system, it will use extreme programming method which consists of planning, designing, coding, and testing. Then, the result that need to be achieved is web-based e-commerce where this system will facilitate the consumer to get information that is needed completely, especially in doing the transaction easily and quickly without the limitation of time and assist the company to extend the broader market by the existence of online selling system.*

*Key Word : E-commerce, PHP, MySQL*

## 1. Latar Belakang

Teknologi mampu memperluas transaksi penjualan dan pembelian produk secara tradisional menuju online, yang sering disebut dengan teknologi *e-commerce*. *E-commerce* adalah penerapan teknologi yang mendukung transaksi penjualan dan pembelian yang dilakukan melalui media elektronik yang di implementasikan dengan sebuah website (D Tri Octafian, 2015).

Perkembangan web saat ini demikian cepat, hal ini disebabkan oleh beberapa faktor diantaranya perkembangan infrastruktur pendukung yang cukup pesat seperti internet, kebutuhan pelaku industri untuk menggunakan web dan internet dalam bisnis mereka.

CV. Sammarindo merupakan badan usaha bergerak dalam bidang produksi sepatu kulit dengan merk Jhon Huss. Sistem pemasaran yang dipakai saat ini masih menggunakan cara konvensional yaitu melalui sms /telepon, *face to face* serta memanfaatkan media sosial sehingga belum menjangkau pangsa pasar secara luas. Selain itu, sistem penjualan produknya juga masih belum terkomputerisasi. Transaksi jual beli kepada pelanggan masih *offline*, pelanggan harus datang langsung untuk memilih ataupun membeli produk yang mereka inginkan.

Berangkat dari permasalahan tersebut, perusahaan memerlukan adanya suatu media untuk penjualan produk serta pemasaran produk melalui *internet*. Untuk mengatasi masalah tersebut, maka dari penulis mengajukan solusi dengan

membuat “ **Rancang Bangun Aplikasi *E-commerce* Pada CV. Sammarindo Berbasis *WEB*** ” dengan menggunakan bahasa pemrograman PHP serta database My SQL.

## 2. Metode Penelitian

Metode penelitian yang digunakan penulis dalam penelitian ini metode kualitatif sebagai berikut:

### 1. Study Literatur

Metode ini dilakukan dengan membaca buku – buku referensi, jurnal yang berhubungan dengan konsep *e-commerce* sebagai landasan dalam memulai penelitian, menganalisis hingga proses perancangan sistem.

### 2. Survey

#### a. Wawancara

Wawancara dilakukan dengan pihak yang terkait untuk mendapatkan informasi mengenai proses bisnis yang sedang berjalan saat ini.

#### b. Observasi

Studi ini dilakukan dengan cara melakukan survei langsung ke perusahaan untuk mendapatkan data serta informasi penting yang dibutuhkan untuk pembuatan sistem. Serta mengamati secara langsung kegiatan yang terjadi pada CV. Sammarindo untuk mengetahui alur kerja dan sistem yang diterapkan.

### 3. Metode Pengembangan Sistem Informasi

Perancangan aplikasi tersebut menggunakan Metode *Extreme Programming* (XP) yaitu sebuah pendekatan pengembangan perangkat lunak yang mencoba meningkatkan efisiensi dan fleksibilitas dari sebuah proyek pengembangan perangkat lunak dengan mengkombinasikan berbagai ide sederhana. Tools Perancangan yang digunakan adalah *Unified Modelling Language* (UML) dalam pengembangan aplikasi dengan empat kerangka kerja (Pressman R. S., 2010), yaitu:

- a) *Planning* (Perencanaan)
- b) *Design* (Perancangan)
- c) *Coding* (Penulisan Kode)
- d) *Testing* (Percobaan)

## 3. LANDASAN TEORI

### 3.1 Pengertian Rancang Bangun

Menurut Kamus Besar Bahasa Indonesia, kata rancang berarti mengatur segala sesuatu sebelum bertindak, mengerjakan atau melakukan sesuatu untuk merencanakan. Sedangkan kata bangun berarti sesuatu yang didirikan. Rancang berarti merencanakan atau mendesain sesuatu yang akan dibuat. Perancangan atau desain didefinisikan sebagai proses aplikasi berbagai teknik dan prinsip bagi tujuan pendefinisian suatu perangkat, suatu proses atau sistem dalam detail yang memadai untuk memungkinkan realisasi fisiknya. Untuk mengendalikan proses desain, A.Davis mengusulkan serangkaian prinsip-prinsip dasar dalam perancangan/desain sebagai berikut:

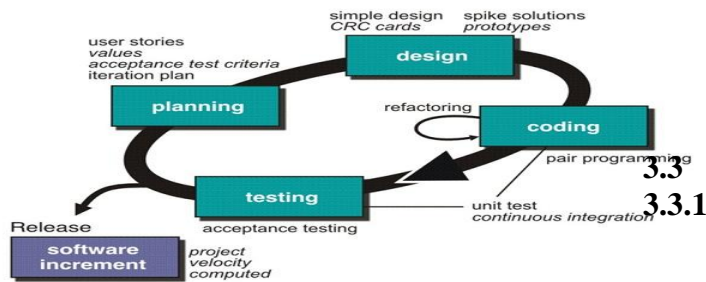
1. Desain tidak boleh berulang.
2. Desain harus terstruktur untuk mengakomodasi perubahan.
3. Desain harus terstruktur untuk berdegradasi dengan baik, bahkan pada saat data dan *event-event* (kejadian-kejadian) menyimpang atau menghadapi kondisi operasi.
4. Desain bukan pengkodean dan pengkodean bukanlah desain.
5. Desain harus dinilai kualitasnya pada saat desain dibuat, bahkan setelah jadi.
6. Desain harus dikaji untuk meminimalkan kesalahan-kesalahan *konseptual* (*semantik*). (Nataniel dkk, 2010 ).

### 3.2 Metode Pengembangan Sistem

#### 3.2.1 *eXtreme Programming*

Menurut (Pressman, 2010), *Extreme Programming* merupakan salah satu dari sekian banyaknya metodologi dalam rekayasa perangkat lunak dan juga merupakan bagian dari metodologi pengembangan perangkat lunak. Secara umum *Extreme Programming* (XP) dapat dijabarkan sebagai sebuah pendekatan pengembangan perangkat lunak yang mencoba meningkatkan efisiensi dan fleksibilitas dari sebuah proyek pengembangan perangkat lunak dengan mengkombinasikan berbagai ide simpel/ sederhana tanpa mengurangi kualitas software yang akan dibangun.

Berikut adalah tahapan - tahapan dalam kerangka kerja *Extreme Programming* :



**Gambar 3.1 Extreme programming**  
**Sumber :Pressman, 2010**

#### 1) Perencanaan (*Planning*)

Kegiatan perencanaan dimulai dengan mendengarkan kegiatan pengumpulan persyaratan yang memungkinkan para anggota teknis tim XP untuk memahami konteks bisnis dari perangkat lunak dan untuk mendapatkan gambaran umum untuk hasil yang diperlukan dan fitur utama dan juga fungsinya.

#### 2) Desain (*Design*)

XP desain secara ketat mengikuti prinsip KIS (keep it simple). Desain sederhana selalu lebih disukai dari pada penjelasan yang lebih kompleks. Selain itu, desain memberikan pedoman pelaksanaan yang mudah dimengerti. Desain adalah fungsi tambahan (karena pengembang menganggap itu akan diperlukan nanti).

#### 3) Coding

Dari pengembangan story dan desain awal, tim tidak pindah ke coding, melainkan mengembangkan serangkaian unit test yang akan dibuat. Setelah uji unit telah dibuat, pengembang lebih fokus pada apa yang harus dilaksanakan untuk menangani permasalahan dalam pengujian.

#### 4) Pengujian (*Testing*)

Pengujian merupakan elemen kunci dari pendekatan XP. Dalam tahap ini terdapat customer tests yang dikhususkan untuk user dan difokuskan pada sistem fitur dan fungsionalitas yang dapat dilihat dan dinilai oleh user. ( Pressman, 2010 )

### **E-Commerce**

#### **Defenisi E-Commerce**

Istilah *e-commerce* mulai muncul di tahun 1990-an melalui adanya inisiatif untuk mengubah paradigma transaksi jual beli dan pembayaran dari cara konvensional ke dalam bentuk digital elektronik berbasis komputer dan jaringan internet. Terdapat beberapa definisi mengenai *e-commerce* seperti berikut ini :

- 1) *E-commerce* menurut (Jony Wong, 2010) merupakan aktivitas pembelian, penjualan dan pemasaran barang atau jasa melalui sistem elektronik.
- 2) Chaffey mendefinisikan *e-commerce* sebagai semua bentuk proses pertukaran informasi antara organisasi dan stakeholder berbasis media elektronik yang terhubung ke jaringan internet. ( Dave Chaffey , 2010 )
- 3) *E-commerce* didefinisikan sebagai proses pembelian dan penjualan produk, jasa dan informasi yang dilakukan secara elektronik dengan memanfaatkan jaringan komputer (Novita Mariana , 2011 )

Jadi dapat disimpulkan bahwa *e-commerce* didefinisikan sebagai proses penjualan dan pembelian produk ataupun jasa serta pertukaran informasi yang dilakukan melalui media elektronik *commerce* yang terhubung dengan jaringan internet.

### 3.4 Sistem berbasis Web

#### 3.4.1 Pengertian Web

“Web adalah salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protokol *HTTP(hypertext transfer protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*. *Browser* (perambah) adalah aplikasi yang mampu menjalankan dokumen-dokumen *web* dengan cara diterjemahkan.” (Rudyanto Arief, 2011).

### 3.5. Pengertian *Unified Modelling Language*(UML)

Membangun model untuk suatu sistem piranti lunak sangat bergantung pada konstruksinya atau kemudahan dalam memperbaikinya. Oleh karena itu, membuat model sangat penting sebagaimana pentingnya memiliki cetak biru untuk bangunan yang besar. Model yang bagus sangat penting untuk menghasilkan komunikasi yang baik antar anggota tim dan untuk meyakinkan sempurnanya arsitektur sistem yang dibangun. Jika ingin membangun suatu model dari suatu sistem yang kompleks, tidak mungkin kita dapat memahaminya secara keseluruhan. Dengan meningkatnya kompleksitas sistem, visualisasi dan pemodelan menjadi sangat penting.

UML dibuat untuk merespon kebutuhan tersebut.

Adi Nugroho (2010) mendefinisikan UML atau *Unified Modeling Language* adalah sebuah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma “berorientasi objek”. Pemodelan sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. (Adi Nugroho 2010).

### 3.6 Pengertian PIECES

Analisis PIECES (*Performance, Information, Economy, control, Eficiency, dan Service*) merupakan teknik untuk mengidentifikasi dan memecahkan permasalahan yang terjadi pada sistem informasi. Dari analisis ini akan menghasilkan identifikasi masalah utama dari suatu sistem serta memberikan solusi dari permasalahan tersebut. Dalam bukunya Hanif Al Fatta tentang *Analisis dan Perancangan Sistem Informasi* dijelaskan bahwa Analisis PIECES terdiri dari:

1. Analisis Kinerja (*Performance*)  
Adalah kemampuan menyelesaikan tugas pelayanan dengan cepat sehingga sasaran atau tujuan segera tercapai. Kinerja diukur dengan jumlah produksi (*Troughput*) dan waktu tanggap (*Respon Time*)

dari suatu sistem. Jumlah Produksi adalah jumlah pekerjaan yang bisa diselesaikan selama jangka waktu tertentu. Sedangkan waktu tanggap adalah waktu transaksi yang terjadi dalam proses kinerja.

## 2. Analisis Informasi (*Information*)

Adalah evaluasi kemampuan sistem informasi dalam menghasilkan nilai atau produk yang bermanfaat untuk menyikapi peluang dalam menangani masalah yang muncul. Situasi dalam analisis informasi ini meliputi:

- Akurasi, informasi harus bebas dari kesalahan dan tidak bias atau menyesatkan.
- Relevan, informasi tersebut memiliki manfaat bagi pihak pemakai maupun pihak pengelola. Dimana relevansi setiap orang berbeda satu dengan yang lainnya.
- Analisis Ekonomi (*Economy*)  
Adalah Penilaian sistem atas biaya dan keuntungan yang akan didapatkan dari sistem yang diterapkan. Sistem ini akan memberikan penghematan operasional dan keuntungan bagi instansi atau perusahaan. Hal yang diperlukan dalam analisis ini meliputi biaya dan keuntungan.
- Analisis Keamanan (*Controlling*)  
Adalah Sistem keamanan yang digunakan harus dapat mengamankan data dari kerusakan, misalnya dengan *memback up data*. Selain itu sistem keamanan juga harus dapat mengamankan data dari akses yang tidak diizinkan. Analisis ini meliputi pengawasan dan pengendalian.
- Analisis Efisiensi (*Efficiency*)  
Adalah sumber daya yang ada guna meminimalkan pemborosan. Efisiensi dari sistem yang dikembangkan adalah pemakaian

secara maksimal terhadap sumberdaya infrastruktur, dan sumberdaya manusia. Serta efisiensi juga menganalisis keterlambatan pengolahan data yang terjadi.

## ➤ Layanan (*Service*)

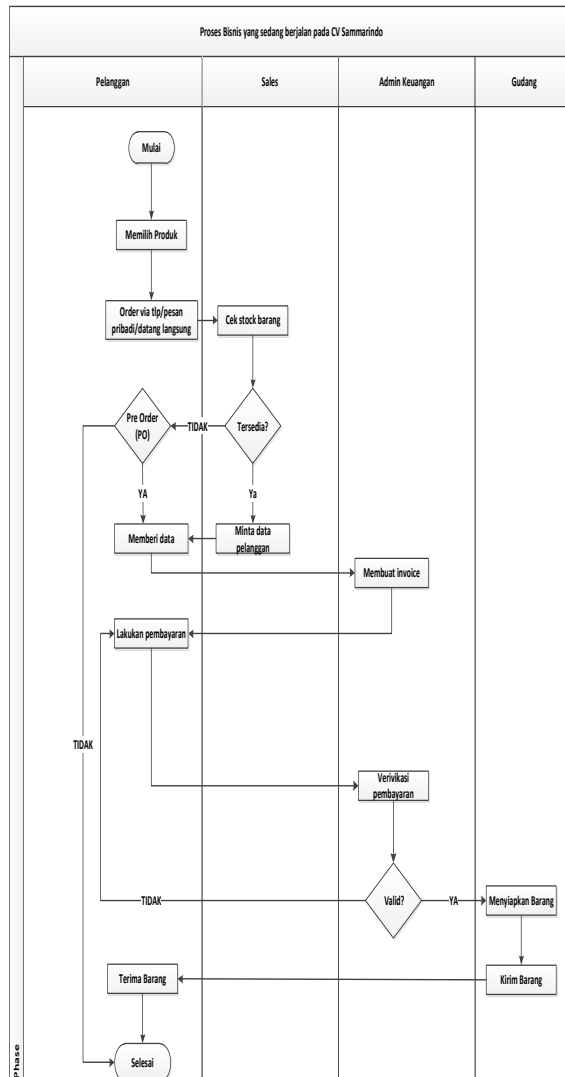
Adalah mengkoordinasikan aktifitas dalam pelayanan yang ingin dicapai sehingga tujuan dan sasaran pelayanan dapat dicapai. (Fatta, dalam jurnal EduLib, Vol 1, No 1 November 2011 ).

## 4. Analisis Masalah

Berdasarkan hasil observasi dan wawancara yang telah dilakukan di CV. Sammarindo, sistem pemesanan dan penjualan produknya masih menggunakan cara tradisional yaitu melalui telepon/sms ,datang langsung dan menggunakan media sosial. Proses bisnis saat ini tergambar pada *flow chart* diatas, dimana transaksi dilakukan secara manual. Hal ini tentunya menjadi kendala baik dalam menjangkau pangsa pasar yang luas maupun proses transaksi penjualan produknya. Sehingga diperlukan suatu sistem penjualan produk berupa aplikasi *e-commerce* berbasis *web* yang dapat memberikan kemudahan kepada pelanggan untuk melakukan transaksi pemesanan maupun pembelian produk yang merekainginkan.

### 4.1 Proses Bisnis Sistem Berjalan

Berikut proses bisnis yang sedang berjalan saat ini pada CV. Sammarindo digambarkan dalam Flow Chart :



**Gambar 4.1 Flowchart Sistem Berjalan**

**Keterangan :**

- 1) Pelanggan memilih barang yang akan dibeli
- 2) Setelah memilih barang yang akan dibeli, pelanggan dapat memesan via telepon, pesan pribadi atau bisa datang langsung.
- 3) Sales akan melakukan pengecekan ke gudang apakah barang tersedia atau tidak. Jika tersedia maka Sales meminta data pelanggan seperti nama, alamat, no telepon.
- 4) Jika barang tidak tersedia maka sales akan menawarkan *pre order* kepada pelanggan. Jika pelanggan bersedia untuk barang PO maka sales akan meminta data pelanggan seperti nama, alamat, no telepon.

- 5) Sales kemudian memberikan data ke bagian admin untuk dibuatkan *invoice*.
- 6) Pelanggan kemudian melakukan pembayaran via transfer ataupun bayar di tempat dan melakukan konfirmasi apabila pembayaran via transfer.
- 7) Admin keuangan verifikasi pembayaran lalu meminta bagian gudang untuk menyiapkan barang sesuai dengan pemesanan.
- 8) Bagian gudang menyiapkan barang sesuai data pemesanan.
- 9) Kemudian bagian gudang mengirimkan barang ke pelanggan.
- 10) Pelanggan menerima barang
- 11) Transaksi selesai.

**4.2 Analisis PIECES Sistem berjalan dan Sistem usulan**

Tabel 4.1 Analisis PIECES Sistem berjalan dan Sistem Usulan

PIECES ANALISIS	SISTEM BERJALAN	SISTEM USULAN
<i>Performance</i>	Kinerja sistem penjualan <i>offline</i> sehingga kurang efektif ketika ingin melakukan transaksi ataupun mendapatkan informasi secara <i>real time</i> . Selain itu promosi produk juga terbatas.	Dengan adanya sistem ini, akan lebih efektif ketika customer melakukan transaksi ataupun ingin produk tanpa terbatas ruang dan waktu.

<b>Information</b>	Informasi penawaran produk masih berupa katalog, dan lewat pesan pribadi. Ketika ada informasi yang ingin disampaikan ke customer harus melalui telepon, sms.	Informasi produk, profil perusahaan, cara bertransaksi tersaji secara <i>online</i> sehingga dapat diakses oleh <i>customer</i> kapan saja dan dimana saja. Selain itu admin juga bias melihat informasi order <i>customer</i> , data pelanggan, serta laporan penjualan.
<b>Economic</b>	Pemborosan kertas karena harus mencatat data order, transaksi dari masing-masing <i>customer</i> . Juga pemborosan biaya pulsa telepon ketika harus menghubungi customer untuk informasi mengenai produk ataupun konfirmasi pembelian.	<i>History order</i> sudah tercatat otomatis saat pelanggan melakukan transaksi, tidak perlu menghubungi <i>customer</i> via telepon saat konfirmasi pembayaran karena sudah otomatis by <i>email</i> dan <i>customer</i> bias cek pada menu tagihan. Selain itu, <i>update</i> informasi produk sudah tersaji pada <i>web</i> dan bias diakses langsung oleh <i>customer</i> .

<b>Control</b>	Sistem yang saat ini berjalan beresiko hilangnya data dan informasi terkait data pelanggan, data pemesanan maupun data laporan penjualan karena hanya dicatat di buku secara manual.	Dokumentasi seluruh data pelanggan, data pemesanan, transaksi serta laporan penjualan sudah terecord pada sistem yang dibangun sehingga data bias lebih terkontrol dengan baik .
<b>Efficiency</b>	Penggunaan sumber daya manusia, informasi, waktu, uang, peralatan, tempat dan keterlambatan pengolahan data order , laporan penjualan tidak bias tersaji dengan cepat saat dibutuhkan.	Dengan system <i>e-commerce</i> ini akan lebih efisien baik dalam penggunaan sumber daya manusia, penyajian informasi, pengolahan data order serta laporan penjualan dapat tersaji lebih cepat dibanding dengan cara manual.



<b>Services</b>	Pelayanan yang diberikan kepada <i>customer</i> kurang maksimal karena system penjualan <i>offline</i> sehingga terbatas dengan waktu dan tempat.	Transaksi secara <i>online</i> meningkat karena <i>customer</i> karena tidak terbatas waktu maupun tempat. Perusahaan dapat memberikan pelayanan yang lebih baik dengan adanya system ini.
-----------------	---------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Tabel 5.1 *requirement* Aktor dan *Use Case*

No	<i>Requirement</i>	Aktor	<i>Use Case</i>
1	Melakukan <i>Login</i>	Semua Aktor	Use Case <i>Login</i>
2	Melakukan daftar member sebagai <i>customer</i> , <i>login</i> , melihat produk <i>ready</i> dan PO, memesan produk, melihat tagihan, melihat profil perusahaan, menulis testimony, update profil.	<i>Customer</i>	<i>Use Case Diagram Customer</i>
3	<i>Login</i> sebagai admin, update data produk, update data produk <i>pre order</i> (PO), update ongkos kirim, view testimony, input data account bank, edit profile.	Admin	<i>Use case Diagram Admin</i>

## 5. Rancangan Usulan

### 5.1 Diagram UML

Perancangan system usulan yang akan dibuat menggunakan pemodelan UML yang merupakan notasi perancangan system berbasis obyek ( *Object Oriented* ). Ada 6 diagram UML yang akan dibuat pada penelitian ini yaitu :

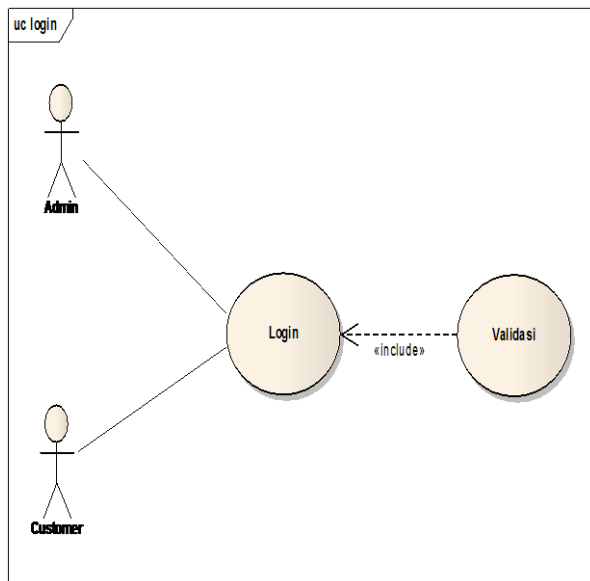
1. *Use Case Diagram*
2. *Activity Diagram*
3. *Sequence Diagram*
4. *Class Diagram*
5. *Component Diagram*
6. *Deployment Diagram*

#### 5.1.1 Use Case Diagram

*Use case diagram* menjelaskan interaksi *actor-actor* yang terlibat dengan sistem yang dibangun serta proses-proses yang ada didalamnya.

Berikut *Requirement* fungsi use case yang dibutuhkan.

### a. Use Case Diagram Login

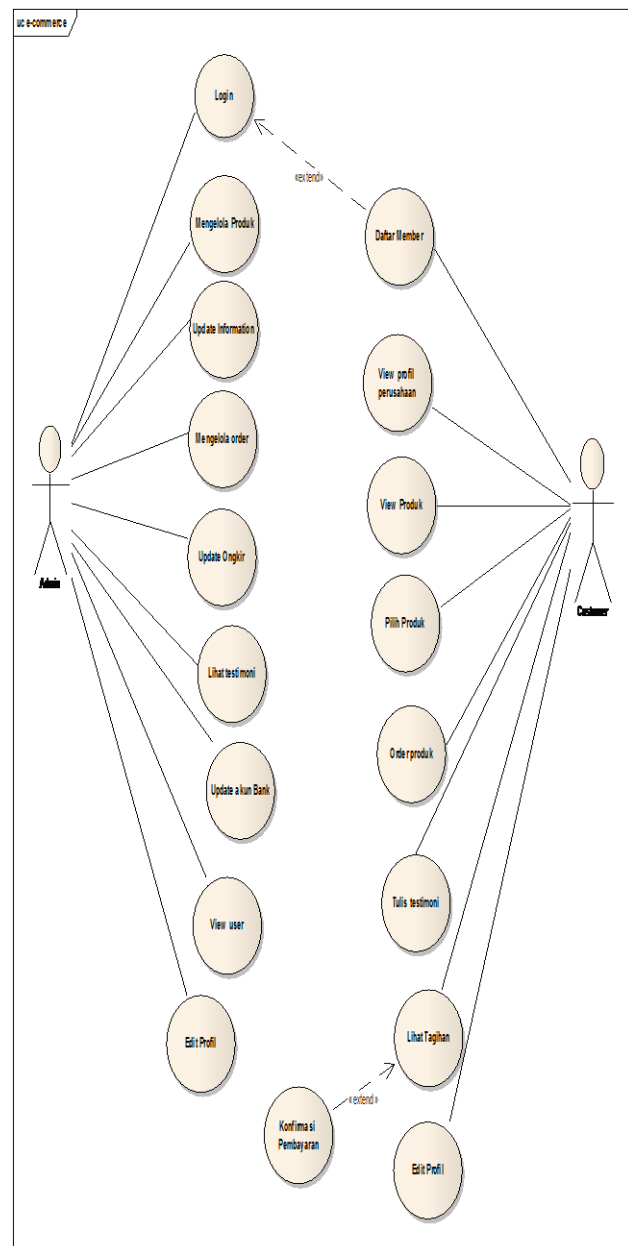


**Gambar 5.1 Use Case Diagram login**

Tabel 5.2 Narrative Use Case Diagram Login

<b>Tujuan</b>	Melakukan Login
<b>Deskripsi</b>	Sistem memungkinkan Actor mengakses area admin dan customer
<b>Actor</b>	Customer, Admin
<b>Scenario Utama</b>	
<b>Kondisi awal</b>	Actor sudah Login ke sistem
<b>Aksi Actor</b>	<b>Reaksi system</b>
Actor melakukan Login	Sistem menampilkan menu utama sesuai dengan hak akses Actor yang Login.
<b>Kondisi Akhir</b>	Actor dapat mengakses menu-menu yang telah ditetapkan oleh sistem.

### a. Use Case Diagram E-Commerce



**Gambar 5.2 Use Case Diagram E-commerce**

Tabel 5.3 Narrative Use Case diagram E-commerce area Admin

<b>Tujuan</b>	Menampilkan area Admin
<b>Deskripsi</b>	Sistem memungkinkan Actor untuk mengakses area admin
<b>Actor</b>	Admin

<i>Scenario utama</i>		<i>Actor</i>	<i>Kegiatan</i>	<i>Deskripsi</i>
<b>Kondisiawal</b>	<i>Actor</i> sudah <i>Login</i> ke sistem			dan menghapus data pengguna
<b>AksiActor</b>	<b>Reaksisystem</b>			
1. <i>Actor login</i> sebagai Admin	Menampilkan menu area Admin		Edit Profil	Melihat dan mengedit profil perusahaan
2. Admin bias mengakses menu-menu yang ada di area admin	Sistem menampilkan halaman yang diakses oleh admin.			

Tabel 5.5 *Narative Use Case diagram e-commerce area Customer*

Tabel 5.4 Deskripsi *Use Case Diagram E-Commerce area Admin*

<i>Actor</i>	<i>Kegiatan</i>	<i>Deskripsi</i>
Admin	<i>Login</i>	<i>Input username dan password</i>
	Mengelola produk	Melihat, menambah, mengedit dan menghapus data produk ready maupun PO
	<i>Update Information</i>	Melihat, menambah, mengedit serta menghapus Informasi seperti Profil perusahaan, cara pemesanan, cara pembayaran.
	Mengelola Order	Melihat, mengupdate, menghapus data transaksi order barang serta menarik laporan penjualan secara periodik
	<i>Update Ongkir</i>	Melihat, mengedit serta menghapus data ongkos kirim paket
	Lihat Testimoni	Melihat testimony dan membalas testimoni customer
	<i>Update akun Bank</i>	Menambah, mengedit dan menghapus akun Bank
	<i>View User</i>	Melihat menambah

<b>Tujuan</b>	Menampilkan area <i>Customer</i>
<b>Deskripsi</b>	Sistem memungkinkan <i>Actor</i> untuk mengakses area <i>Customer</i>
<b>Actor</b>	<i>Customer</i>
<b>Scenario utama</b>	
<b>Kondisiawal</b>	<i>Actor</i> sudah <i>Login</i> ke sistem
<b>AksiActor</b>	<b>Reaksisystem</b>
1. <i>Actor login</i> sebagai <i>Member</i>	Menampilkan menu area <i>Customer</i>
2. Admin bisa mengakses menu-menu yang ada di area <i>Customer</i>	Sistem menampilkan halaman yang diakses oleh <i>Customer</i>

Tabel 5.6 Deskripsi *Use Case Diagram e-commerce area Customer*

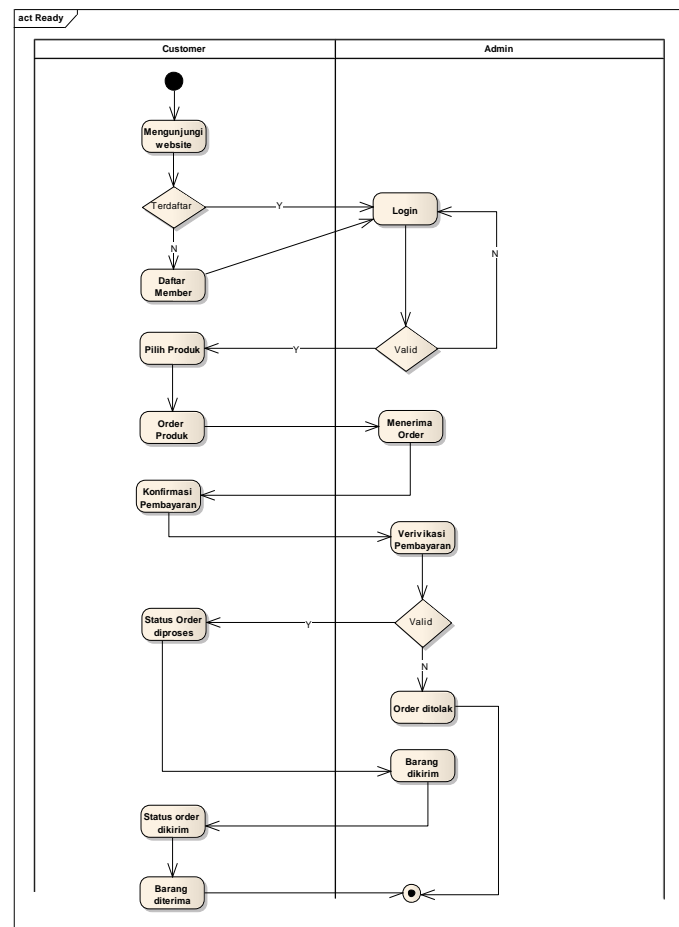
<i>Actor</i>	<i>Kegiatan</i>	<i>Deskripsi</i>
	Daftar Member	<i>Customer</i> melakukan pendaftaran sebagai member pada menu daftar
	<i>Login</i>	<i>Login</i> dengan memasukan <i>username</i> dan <i>password</i> sesuai dengan <i>username</i> dan

Actor	Kegiatan	Deskripsi
Customer		password saat mendaftar sebagai member
	View Produk	Melihat produk-produk yang ditawarkan pada <i>website</i> .
	View Profil Perusahaan	Melihat Profil Perusahaan
	Pilih Produk	Customer dapat memilih produk yang diinginkan
	Order Produk	Customer dapat melakukan order produk yang tersedia maupun produk po dengan cara klik <i>add to cart</i> .
	Tulis testimoni	Customer dapat melihat, menulis serta menghapus testimoni
	Lihat Tagihan	Untuk melihat tagihan belanja, melakukan konfirmasi pembayaran serta melihat status pemesanan dapat dilihat pada menu tagihan anda
	Edit Profil	Customer mengupdate profil jika ada perubahan dari data yang sebelumnya

### 5.1.2 Activity Diagram

Berikut diagram *workflow* yang menggambarkan tentang aktivitas dari ketika melakukan setiap kegiatan berdasarkan *use case* sistem usulan :

### A. Activity Diagram Penjualan Produk Ready Stock



**Gambar 5.3 Activity Diagram Penjualan Produk Ready Stock**

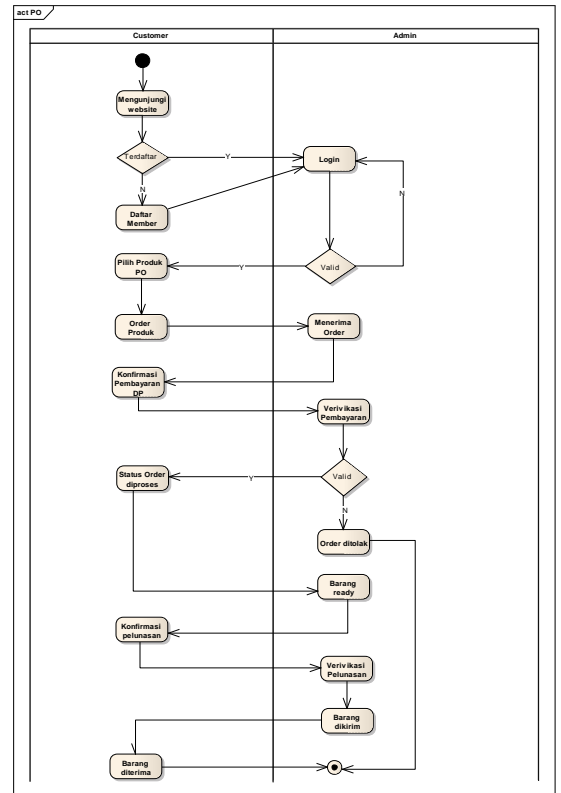
### Keterangan :

- 1) User mengunjungi *website*.
- 2) Jika *user* sudah terdaftar menjadi member maka *user* dapat melakukan *login* sebagai member.
- 3) Sebaliknya, jika *user* belum terdaftar sebagai member maka harus melakukan pendaftaran terlebih dahulu .
- 4) User melakukan *login* sebagai *member/admin*.
- 5) Jika *username* dan *password* sudah *valid* maka *member* akan masuk ke halaman *web* sesuai

dengan hak akses ( sebagai *customer/ admin*).

- 6) Jika *username* dan *password* tidak *valid* maka harus melakukan *login* kembali dengan memasukkan *username* dan *password* yang benar.
- 7) *Customer* memilih produk yang ditawarkan pada *website*
- 8) *Customer* melakukan order produk sesuai keinginan
- 9) Admin menerima order dari *customer*
- 10) *Customer* menerima tagihan dan melakukan konfirmasi pembayaran.
- 11) Admin melakukan verifikasi pembayaran dari *customer*.
- 12) Jika pembayaran sudah sesuai maka barang dikirim .
- 13) Status order di *customer* *done* dan menunggu sampai barang diterima.

## B. Activity Diagram Transaksi Pre Order



**Gambar 5.4 Activity Diagram Transaksi Pre Order**

Berikut penjelasan Activity diagram transaksi *pre order* pada gambar 4.5 :

1. User mengunjungi *website*.
2. Jika *user* sudah terdaftar menjadi member maka *user* dapat melakukan *login* sebagai member.
3. Sebaliknya, jika *user* belum terdaftar sebagai member maka harus melakukan pendaftaran terlebih dahulu .
4. User melakukan *login* sebagai *member/admin*.
5. Jika *username* dan *password* sudah *valid* maka *member* akan masuk ke halaman *web* sesuai dengan hak akses ( sebagai *customer/ admin*).
6. Jika *username* dan *password* tidak *valid* maka harus

7. *Customer* memilih produk pre order (PO) jika barang tidak *ready stock*.
8. *Customer* melakukan order (pemesanan) produk pre order.
9. Admin menerima order dari *customer*.
10. *Customer* menerima konfirmasi pembayaran DP.
11. Admin melakukan verifikasi pembayaran DP.
12. Jika pembayaran DP sudah sesuai maka order *customer* akan diproses.
13. Setelah barang *ready* admin akan konfirmasi pelunasan kepada *customer*.
14. Jika *Customer* sudah melakukan pembayaran, *customer* konfirmasi pelunasan DP pada menu tagihan
15. Admin melakukan *verifikasi* pelunasan sisa pembayaran
16. Barang pesanan siap dikirim
17. Status *order* selesai, dan *customer* menunggu sampai barang diterima.

[illegible]

**Gambar 5.5 Class Diagram Sistem Usulan**

**Keterangan :**

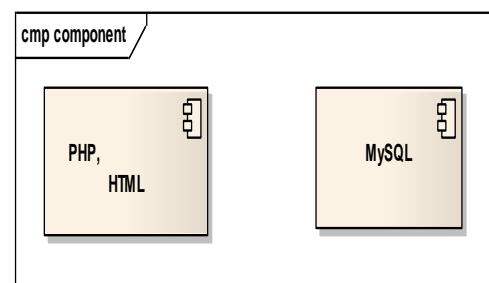
1. Satu user dapat mengelola banyak informasi.
2. Satu user dapat membuat banyak testimoni.

3. Satu user dapat *reply* testimoni.
4. Satu user mengelola banyak data akun bank.
5. Satu user mengelola banyak tarif.
6. Satu user mengelola banyak produk
7. Satu user mengelola banyak transaksi.
8. Satu user mengelola satu site konfigurasi.
9. Satu kategori memiliki banyak produk.
10. Satu produk stock memiliki satu produk.
11. Satu produk memiliki banyak shopping chart.
12. Satu produk memiliki banyak transaksi.
13. Satu transaksi detail memiliki satu transaksi.
14. Satu kurir paket memiliki banyak tarif.
15. Satu kurir memiliki satu kurir paket.
16. Satu kota memiliki satu tarif.
17. Satu kota memiliki banyak kecamatan.

#### 5.1.4 Component Diagram

*Component* diagram adalah diagram UML yang menampilkan komponen dalam system dan hubungan antara mereka. Pada

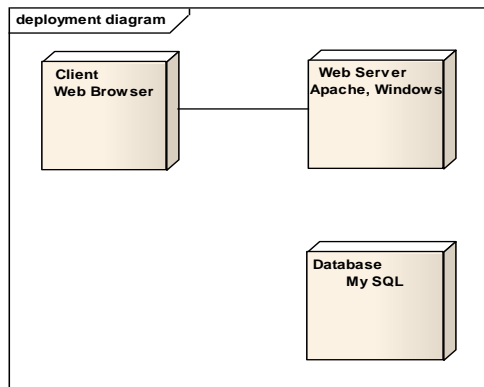
*component View*, akan difokuskan pada organisasi fisik system. Pertama, diputuskan bagaimana kelas-kelas akan diorganisasikan menjadi kode pustaka. Kemudian akan dilihat bagaimana perbedaan antara berkas eksekusi, berkas dynamic link library (DDL), dan berkas runtime lainnya dalam system. *Component* diagram mempresentasikan dunia real item yaitu component software. Component software adalah bagian fisik dari sebuah sistem karena menetap di komputer. Component diagram mengandung component, interface dan relationship (Munawar, 2010).



**Gambar 5.6 Component Diagram**

#### 5.1.5 Deployment Diagram

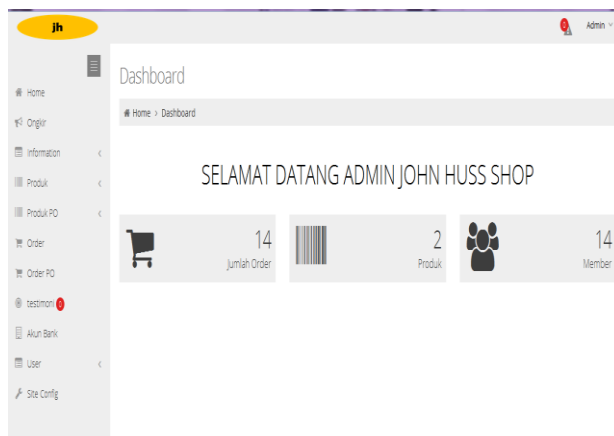
*Deployment* diagram menunjukkan tata letak sebuah sistem secara fisik, menampilkan bagian-bagian software yang berjalan pada bagian-bagian hardware (Munawar, 2010).



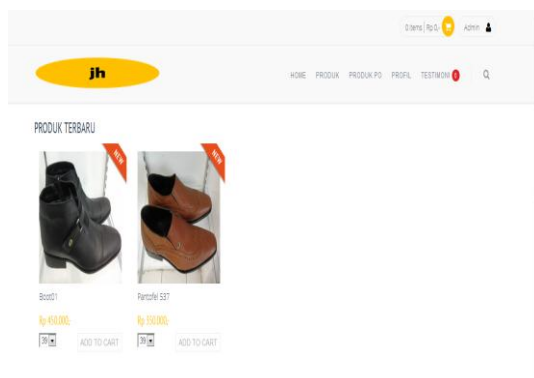
**Gambar 5.7 Deployment Diagram**

## 6. Struktur Navigasi Sistem Usulan

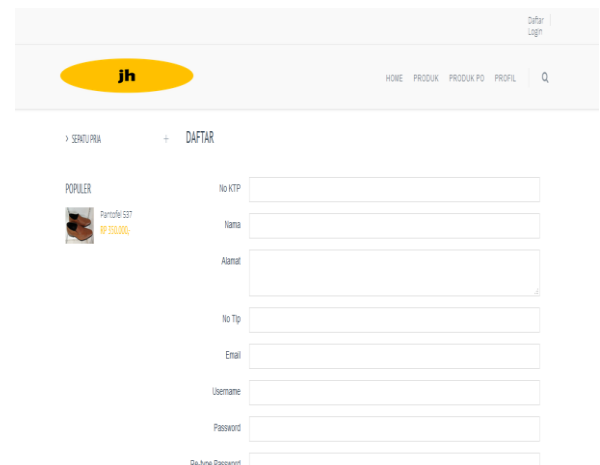
### 6.1 Tampilan Halaman Utama Admin



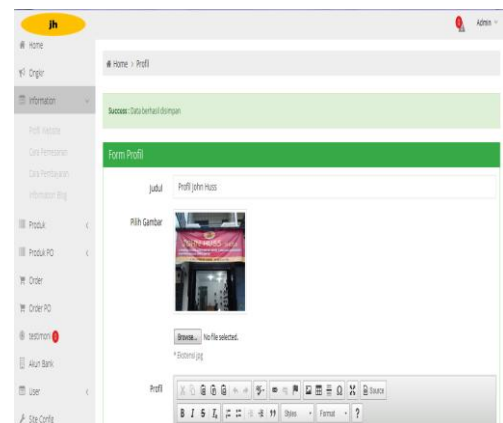
### 6.2 Tampilan Halaman Utama User



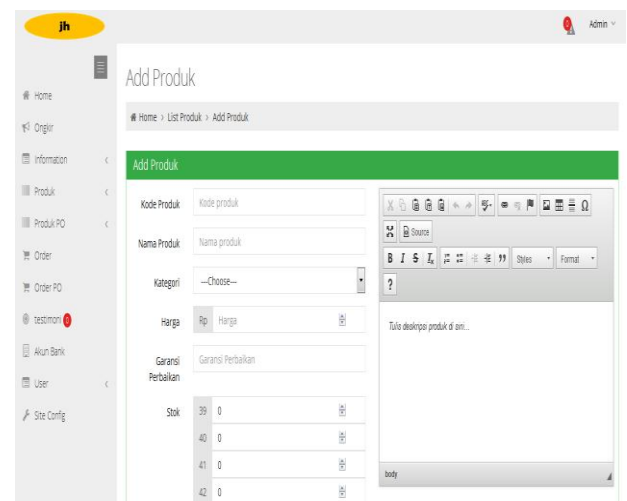
### 6.3 Tampilan Halaman Utama User



### 6.4 Tampilan Halaman Update Profile Perusahaan



### 6.5 Tampilan Halaman Tambah Produk





## 6.5 Tampilan Halaman Tambah Kategori Produk

The screenshot shows the 'Tambah Kategori' page. On the left, there is a form with a 'Parent' dropdown set to 'Root', a 'Nama Kategori' input field with the placeholder 'Masukkan Nama Kategori', and a 'Save' button. On the right, there is a table listing existing categories:

Nama Kategori	Action
Sepatu Pria	<a href="#">Edit</a> <a href="#">Hapus</a>
- Pantofel	<a href="#">Edit</a> <a href="#">Hapus</a>
- Boot	<a href="#">Edit</a> <a href="#">Hapus</a>
- Semi Boot	<a href="#">Edit</a> <a href="#">Hapus</a>
- Jingle	<a href="#">Edit</a> <a href="#">Hapus</a>

## 6.6 Tampilan Halaman Tambah Produk PO

The screenshot shows the 'Add Produk PO' page. The form includes fields for 'Kode Produk PO', 'Nama Produk', 'Kategori' (a dropdown menu), 'Harga' (with a currency selector set to 'Rp'), 'Garansi Perbaikan', 'Berat Kotak (kg)', and 'Warna'. There is also a rich text editor for 'Tulis deskripsi produk di sini...'.

## 6.7 Tampilan Halaman Tambah Kategori Produk PO

The screenshot shows the 'Tambah Kategori PO' page. On the left, there is a form with a 'Parent' dropdown set to 'Root', a 'Nama Kategori PO' input field with the placeholder 'Masukkan Nama Kategori', and a 'Save' button. On the right, there is a table listing existing categories:

Nama Kategori PO	Action
Sepatu Pria	<a href="#">Edit</a> <a href="#">Hapus</a>
- Pantofel	<a href="#">Edit</a> <a href="#">Hapus</a>

## 6.8 Tampilan Halaman Lihat Detail Order

The screenshot shows the 'View Order' page. It displays order details on the left and customer information on the right.

Order Detail	Customer Information
Purchase Order : PO014	No Ktp :
Tanggal Order : 05 Agustus 2016	Nama : Admin
Total Harga : Rp 92.000,-	No. Tlp : 857-2023-7957
Tanggal Transfer : 05 Agustus 2016	Username : admin
Total Transfer : Rp 92.000,-	<b>PENGERIMAN</b>
Bukti Transfer : <a href="#">Lihat</a>	Nama : bibi
Tanggal Kirim : 05 Agustus 2016	Kota / Kabupaten : Kab Situbondo
Nomor Resi : 213213	Kecamatan : Banyuwangi
	Alamat : aaaaaaa

# 7. KESIMPULAN DAN SARAN

## 7.1 Kesimpulan

Berdasarkan uraian yang telah dibahas pada bab-bab sebelumnya, maka penulis membuat kesimpulan sebagai berikut :

1. Sistem yang diusulkan berbasis web dimana *customer* dapat melakukan transaksi secara *online*.

2. Dengan adanya aplikasi *e-commerce* ini, jangkauan pasar untuk penjualan produk akan lebih luas.
3. Seluruh data pelanggan, pemesanan dan pembayaran tersimpan dalam database.

## 7.2 SARAN

Ada beberapa saran sebagai *improvement* aplikasi *e-commerce* yang sudah dibuat yaitu :

1. Member dibedakan menjadi 2 (dua) untuk membedakan *customer* yang membeli satuan dengan *customer* yang membeli grosiran.
2. Transaksi pembayaran saat ini hanya via transfer, jadi disarankan untuk penambahan transaksi pembayaran seperti *e-pay*.

## Daftar Pustaka

- Arief, M.Rudiyanto. 2011. Pemograman WEB Dinamis Menggunakan PHP & My SQL. Bandung :Penerbit Andi
- Chaffey, Dave. 2010. *E-Business and E-Commerce Management*, 4thEdition, Prentice Hall, Inc. New Jersey. Corones, Stephen. Market ,*Publisher Chengage Learning*, Boston
- D Tri Octafian. Jan 2015. WEB Multi *E-commerce* berbasis Framework Codeigniter, Jurnal Teknologi dan Informatika(TEKNOMATIKA) VOL. 5 NO. 1
- Fatta, Al. Analisis Sistem Informasi Perpustakaan IAIN Surakarta, Jurnal Edulib, Vol 1 November 2011 ).
- Mariana, Novita dkk. Januari 2011. *Ecommerce* Sentra Kluster IKM Sepatudi Kota Semarang sebagai Media Promosi dan Komunikasi OnLine Jurnal Teknologi Informasi DINAMIK volume16,No 1.
- Nataniel & Heliza Rahmania Hatta. Feb 2009. Perancangan Sistem Informasi Terpadu Pemerintah Daerah Kabupaten Paser, Jurnal Informatika Mulawarman Vol 4.1
- Nugroho, Adi . 2010. Analisa dan Perancangan Sistem Informasi dengan metologi Berorientasi Objek. Bandung: Penerbit Informatika
- Pressman, R. S., 2010. Software Engineering: A Practitioner's Approach, Seventh Edition. In: New York: McGraww-Hill, New York .
- Wong, Jonny. 2010. Internet Marketing for Beginners, Elex Media Komputindo, Jakarta.