

Menu Dan Panduan Penggunaan

RANCANG BANGUN PROGRAM PENJUALAN TAS KULIT PADA UKM

(Studi Kasus pada Exotic Cikutra Bandung)

1.1 Pendahuluan

Usaha Kecil Menengah atau yang sering disingkat UKM merupakan salah satu bagian penting dari perekonomian suatu negara maupun daerah, begitu juga dengan negara Indonesia. UKM ini sangat memiliki peranan penting dalam lajunya perekonomian masyarakat. UKM ini juga sangat membantu negara atau pemerintah dalam hal penciptaan lapangan kerja baru dan lewat UKM juga banyak tercipta unit-unit kerja baru yang menggunakan tenaga-tenaga baru yang dapat mendukung pendapatan rumah tangga. Selain dari itu UKM juga memiliki fleksibilitas yang tinggi jika dibandingkan dengan usaha yang berkapasitas lebih besar. UKM ini perlu perhatian yang khusus dan di dukung oleh informasi yang akurat, agar terjadi link bisnis yang terarah antara pelaku usaha kecil dan menengah dengan elemen daya saing usaha, yaitu jaringan pasar. Pada tataran perusahaan yang bergerak dibidang penjualan membutuhkan suatu sistem informasi yang baik, terutama sistem informasi penjualan, agar dalam kegiatannya dapat berjalan dengan efektif dan efisien. Keadaan tersebut menyebabkan banyaknya perusahaan yang meningkatkan pengembangan dibidang penjualan untuk meningkatkan pelayanan yang lebih baik lagi dan dapat mengolah data dengan mudah, cepat dan akurat. Demikian juga halnya yang terjadi dengan Toko Tas House Of Leather, sebagian besar proses pengolahan data dan penyajian informasinya masih secara manual. yang dapat menimbulkan kelemahan, misalnya membutuhkan waktu yang lama dalam pemasukan data, pencarian data, pembuatan, dan penyajian laporan, atau dengan kata lain tidak efisien dalam menggunakan waktu kerja. Faktor dominan yaitu kesalahan manusia banyak terjadi dalam mengolah data serta penyimpanan dan perawatan dokumen berbentuk kertas, dimana hal ini sangat sulit ditanggulangi. Hal tersebut mendorong penulis untuk mengatasi permasalahan-permasalahan yang ada saat ini, yaitu dibutuhkan suatu aplikasi berbentuk pemrograman yang diharapkan dapat menjadi tempat baik bagi pelanggan maupun untuk UKM itu sendiri dalam menjual dan mencari produk yang dibutuhkan.

Dengan menggunakan Microsoft Visual basic 6.0, penulis tertarik untuk menciptakan suatu program basis data penjualan. Dengan digunakannya perancangan basis data penjualan dan persediaan barang diharapkan mampu menjawab kelemahan-kelemahan yang selama ini muncul.

1.2 Identifikasi Masalah

1. Perhitungan transaksi penjualan dan pembayaran masih dilakukan secara manual di tulis di kertas (manual book)
2. Pemeriksaan stok barang akan memakan waktu yang lama karena harus mengecek barang-barangnya satu persatu
3. Pembuatan laporan penjualan seringkali mengalami kesalahan karena dilakukan dengan cara menyalin kembali nota-nota yang di kumpulkan oleh petugas penjualan.

1.3 Rumusan Masalah

Berdasarkan latar belakang diatas, terdapat permasalahan yang timbul sebelum dibuatnya perancangan aplikasi penjualan yaitu:

1. Bagaimana membuat suatu sistem aplikasi penjualan yang sesuai dengan kebutuhan sehingga memudahkan karyawan dalam mengolah data , transaksi dan pembuatan laporan penjualan, serta bagi pimpinan mempermudah dalam pengambilan keputusan.

1.4 Batasan Masalah

Karena luasnya permasalahan penyusunan dalam Tugas Akhir ini, maka penulis memberi batasan masalah dan asumsi yang dibahas dalam Tugas akhir ini meliputi :

1. Aplikasi ini hanya membahas tentang Transaksi penjualan yang ada di toko Exotic Cikutra Bandung saja.
2. Pengembangan sistem menggunakan metoda waterfall yang dilakukan tidak sampai maintenance.

1.5 Maksud dan Tujuan Penelitian

1.5.1 Maksud Penelitian

Maksud penelitian ini adalah untuk merancang dan membuat suatu Aplikasi Penjualan, sehingga pimpinan dan petugas penjualan dapat mengelola data yang sudah tersedia dan tersimpan menyatu di dalam satu aplikasi ini.

1.5.2 Tujuan Penelitian

membuat sistem aplikasi penjualan yang sesuai dengan kebutuhan sehingga memudahkan karyawan dalam mengolah data , transaksi dan pembuatan laporan penjualan, serta bagi pimpinan mempermudah dalam pengambilan keputusan

1.6 Metodologi Penelitian

Dalam melakukan penyusunan laporan ini menggunakan metode penelitian sebagai berikut:

1. Studi Lapangan

Yaitu dengan melakukan observasi dalam penelitian langsung ke objek yang diteliti.

2. Wawancara

Yaitu teknik dengan mengajukan pertanyaan-pertanyaan yang bersangkutan dengan penelitian sebagai bahan kaji.

3. Studi Dokumentasi

Yaitu dengan melihat dan mempelajari dokumen yang berhubungan dengan data kajian

4. Riset Kepustakaan

Yaitu dengan membaca dan mempelajari buku-buku yang berhubungan dengan sistem informasi.

2.1 Landasan Teori

2.1 Pengertian Program

Pengertian Program menurut *Jogiyanto*, adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.

Menurut Kamus *Kamus Besar Bahasa Indonesia* “Program adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu”[6].

Program adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Aplikasi merupakan rangkaian kegiatan atau perintah untuk dieksekusi oleh komputer. [8]

2.2 Pengertian Penjualan

Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga dapat mengetahui hasil produk yang dihasilkan. Penjualan dalam lingkup kegiatan, sering disalah artikan dengan pengertian pemasaran. Penjualan dalam lingkup ini lebih berarti tindakan menjual barang atau jasa.

Penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencana-rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba. (Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga dapat mengetahui hasil produk yang dihasilkan.

2.3 Object Oriented Programming (OOP)

Object Oriented Programming (OOP) adalah suatu metode pemrograman yang berbasiskan pada objek, secara singkat pengertian dari OOP adalah koleksi objek yang saling berinteraksi dan saling memberikan informasi satu dengan yang lainnya[7]. Suatu program disebut dengan pemrograman berbasis objek (OOP) karena terdapat :

1. Encapsulation (pembungkusan)

- a. Variabel dan method dalam suatu objek dibungkus agar terlindungi
- b. Untuk mengakses, variabel dan method yang sudah dibungkus tadi perlu interface
- c. Setelah variabel dan method dibungkus, hak akses terhadapnya bisa ditentukan.
- d. Konsep pembungkusan ini pada dasarnya merupakan perluasan dari tipe data struktur

2. Inheritance (pewarisan)

- a. Sebuah class bisa mewariskan atribut dan method-nya ke class yang lain
- b. Class yang mewarisi disebut superclass
- c. Class yang diberi warisan disebut subclass
- d. sebuah subclass bisa mewariskan atau berlaku sebagai superclass bagi class yang lain disebut $\longrightarrow$ *multilevel inheritance*. Keuntungan Penggunaan Pewarisan
- e. Subclass memiliki atribut dan method yang spesifik yang membedakannya dengan superclass, meskipun keduanya mirip (dalam hal kesamaan atribut dan method).
- f. Dengan demikian pada pembuatan subclass, programmer bisa menggunakan ulang source code dari superclass yang ada ini yang disebut dengan istilah *reuse*.
- g. Class-class yang didefinisikan dengan atribut dan method yang bersifat umum yang berlaku baik pada superclass maupun subclass disebut dengan abstract class.

3. Polymorphism (polimorfisme – perbedaan bentuk)

Polimorfisme artinya penyamaran dimana suatu bentuk dapat memiliki lebih dari satu bentuk.

2.4 UML

Unified Modeling Language (UML) adalah “keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO).”[4]

Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corp. UML menyediakan notasi- notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

UML dideskripsikan oleh beberapa diagram, diantaranya:

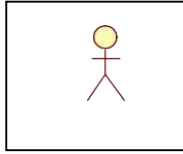
1. Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case diagram* lebih dititik beratkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Sebuah *use case diagram* merepresentasikan sebuah interaksi antara aktor dengan sistem.

Komponen-komponen yang terlibat dalam *use case diagram* :

a. Aktor

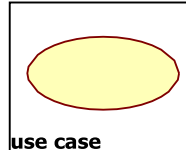
Pada dasarnya aktor bukanlah bagian dari *use case diagram*, namun untuk dapat terciptanya suatu *use case diagram* diperlukan aktor, dimana aktor tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat atau sistem lain) yang berinteraksi dengan sistem yang dibuat. Sebuah aktor mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima dan memberi informasi pada sistem. Aktor hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*. Aktor digambarkan dengan *stick man*.



Gambar 2.1 Aktor

b. Use Case

Gambaran fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti kegunaan sistem yang akan dibangun.



Gambar 2.2 Use Case

Ada beberapa relasi yang terdapat pada *use case* diagram:

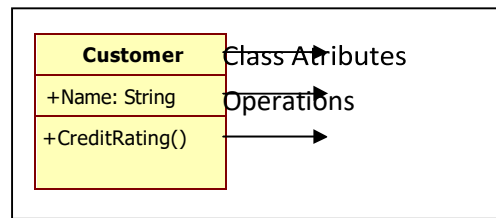
1. *Association*, menghubungkan link antar element.
2. *Generalization*, disebut juga pewarisan (*inheritance*), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. *Dependency*, sebuah element bergantung dalam beberapa cara ke element lainnya.
4. *Aggregation*, bentuk *association* dimana sebuah elemen berisi elemen lainnya.

Tipe relasi yang mungkin terjadi pada *use case* diagram:

1. `<<include>>`, yaitu Relasi usecase tambahan ke sebuah usecase dimana usecase yang di tambahkan memerlukan usecase ini untuk menjalankan.
2. `<<extends>>`, Relasi usecase tambahan ke sebuah usecase dimana usecase yang di tambahkan dapat berdiri sendiri walau tanpa usecase tambahan itu.
3. `<<communicates>>`, merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara aktor dan *use case*.

c. Class Diagram

Class adalah sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain.



Gambar 2.3 Class Diagram

Class memiliki tiga area pokok :

1. Nama (*Class Name*)
2. Atribut
3. Metode (*Operation*)

Pada UML, *class* digambarkan dengan segi empat yang dibagi beberapa bagian. Bagian atas merupakan nama dari *class*. Bagian yang tengah merupakan struktur dari *class* (atribut) dan bagian bawah merupakan sifat dari *class* (metode/operasi).

Atribut dan metode dapat memiliki salah satu sifat berikut :

1. *Private* , tidak dapat dipanggil dari luar *class* yang bersangkutan.
2. *Protected* , hanya dapat dipanggil oleh *class* yang bersangkutan dan *class* lain yang mewarisinya.
3. *Public* , dapat dipanggil oleh *class* lain.

Hubungan antar *Class* :

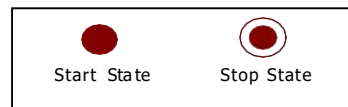
1. Asosiasi, yaitu hubungan statis antar *class*. Umumnya menggambarkan *class* yang memiliki atribut berupa *class* lain, atau *class* yang harus mengetahui eksistensi *class* lain.

2. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas”).
3. Pewarisan, yaitu hubungan hirarki antar *class*. *Class* dapat diturunkan dari *class* lain dan mewarisi semua atribut dan metode *class* asalnya serta bisa menambahkan fungsionalitas baru. Sehingga *class* tersebut disebut anak dari *class* yang diwarisinya.
4. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *class* kepada *class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *sequence diagram* yang akan dijelaskan kemudian.

d. Statechart Diagram

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu objek dari suatu *class* dan keadaan yang menyebabkan *state* berubah. *Statechart* diagram tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah oleh *state* yang berbeda.

State adalah sebuah kondisi selama kehidupan sebuah objek atau ketika objek memenuhi beberapa kondisi, melakukan beberapa aksi atau menunggu sebuah *event*. *State* dari sebuah objek dapat dikarakteristikkan oleh nilai dari satu atau lebih atribut-atribut dari *class*. *State* dari sebuah objek ditemukan dengan pengujian/pemeriksaan pada atribut dan hubungan dari objek. Notasi UML untuk *state* adalah persegi panjang/bujur sangkar dengan ujung yang dibulatkan.



Gambar 2.4 Start State dan Stop State

Masing-masing diagram harus mempunyai satu dan hanya satu *start state*

ketika objek mulai dibuat. Sebuah objek boleh mempunyai banyak *stop state*.



Gambar 2.5 State Transition

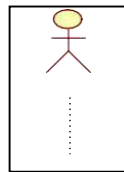
Sebuah *state transition* dapat mempunyai sebuah aksi dan/atau sebuah kondisi penjaga (*guard condition*) yang terasosiasi dengannya, dan mungkin juga memunculkan sebuah *event*. Sebuah aksi adalah kelakuan yang terjadi ketika *state transition* terjadi. Sebuah *event* adalah pesan yang dikirim ke objek lain di sistem. Kondisi penjaga adalah ekspresi *boolean* (pilihan Ya atau Tidak) dari nilai atribut- atribut yang mengijinkan sebuah *state transition* hanya jika kondisinya benar. Kedua aksi dan penjaga adalah kelakuan dari objek dan secara tipikal menjadi operasi.

2.2 Linier Sequential Model

e. *Sequence Diagram*

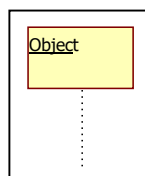
Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

Dibawah merupakan simbol yang digunakan pada *sequence diagram* :



Gambar 2.6 Aktor

Actor adalah pesan dari seseorang atau sistem lain yang bertukar informasi dengan sistem yang lainnya, kemudian lifeline berhenti atau mulai pada titik yang tepat.



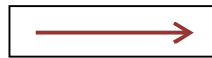
Gambar 2.7 *Object lifeline*

Object lifeline menunjukkan keberadaan dari sebuah objek terhadap waktu. Yaitu objek dibuat atau dihilangkan selama suatu periode waktu diagram ditampilkan, kemudian lifeline berhenti atau mulai pada titik yang tepat.



Gambar 2.8 *Activation*

Activation menampilkan periode waktu selama sebuah objek atau aktor melakukan aksi. Dalam *object lifeline*, *activation* berada diatas *lifeline* dalam bentuk kotak persegi panjang, bagian atas dari kotak merupakan inisialisasi waktu dimulainya suatu kegiatan dan yang dibawah merupakan akhir dari waktu.

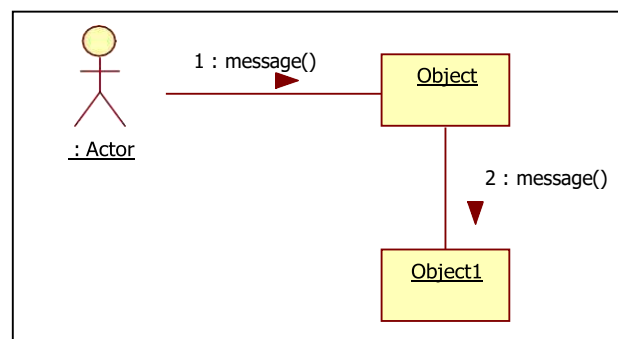


Gambar 2.9 *Message*

Message adalah komunikasi antar objek yang membawa informasi dan hasil pada sebuah aksi. *Message* menyampaikan dari *lifeline* sebuah objek kepada *lifeline* yang lain, kecuali pada kasus sebuah *message* dari objek kepada objek itusendiri, atau dengan kata lain *message* dimulai dan berakhir pada *lifeline* yang sama.

f. Collaboration Diagram

Diagram ini menggambarkan interaksi objek yang diatur objek sekelilingnya dan hubungan antara setiap objek dengan objek yang lainnya. Dalam menunjukkan pertukaran pesan, *collaboration diagram* menggambarkan objek dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan gunakan *sequence diagram*, tapi jika penekanannya pada konteks gunakan *collaboration diagram*.

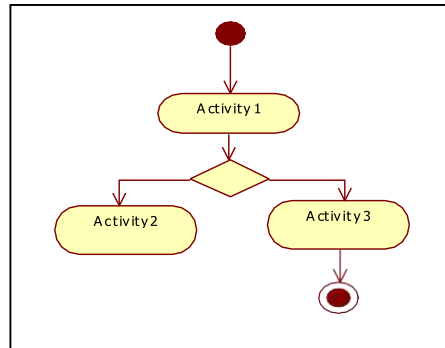


Gambar 2.10 *Collaboration Diagram*

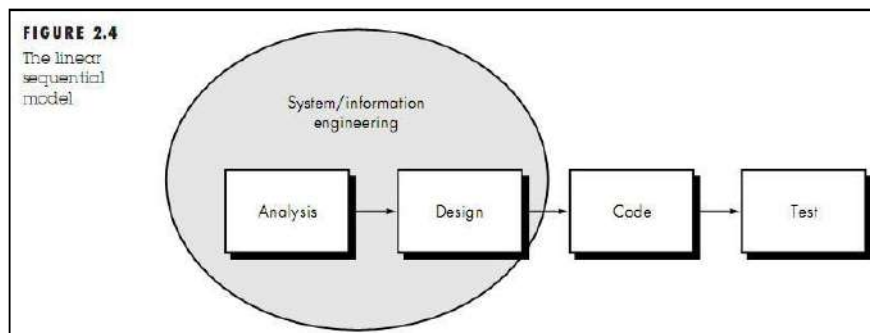
g. Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari satu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Pembuatan *activity diagram* pada awal pemodelan proses dapat membantu memahami keseluruhan proses. *Activity diagram* juga digunakan untuk menggambarkan interaksi antara beberapa *use case*.

Gambar 2.11 Activity Diagram



Terkadang disebut *Classic Life Cycle* atau *Waterfall Model*, yaitu yang mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak yang sistematis dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh *analisis*, *desain*, *kode*, *pengujian*, dan *pemeliharaan*. Dimodelkan setelah siklus rekayasa konvensional, model sekuensial linier melingkupi aktivitas- aktivitas sebagai berikut :[3]



Gambar 2.12 Pemodelan Linier Sequential atau Waterfall

a. Rekayasa dan Pemodelan Sistem/ Informasi (*System/ Information Engineering and Modeling*).

Karena perangkat lunak selalu merupakan bagian dari sebuah sistem (bisnis) yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan

beberapa subset dari kebutuhan ke perangkat lunak tersebut. Pandangan sistem ini penting ketika perangkat lunak harus berhubungan dengan elemen-elemen yang lain seperti perangkat lunak, manusia, dan database. Rekayasa dan analisis sistem menyangkut pengumpulan kebutuhan pada tingkat sistem dengan jumlah kecil analisis serta desain tingkat puncak. Rekayasa informasi mencakup juga pengumpulankebutuhan pada tingkat bisnis strategis dan tingkat area bisnis.

b. Analisa Kebutuhan Perangkat Lunak (*Software Requirements Analysis*)

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khususnya pada perangkat lunak. Untuk memahami sifat program yang dibangun, perekraya perangkat lunak (analisis) harus memahami domain informasi, tingkah laku, unjuk kerja, dan antar muka (*interface*) yang diperlukan. Kebutuhan baik untuk sistem maupun perangkat lunak didokumentasikan dan dilihat lagi dengan pelanggan.

c. Desain (*Design*)

Desain perangkat lunak sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda :

Struktur data, arsitektur perangkat lunak, representasi interface, dan detail(algoritma) prosedural. Proses desain menerjemahkan syarat/ kebutuhan dalam sebuah representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode. Sebagaimana persyaratan, desain didokumentasi dan menjadi bagian dari konfigurasi perangkat lunak.

d. Pengkodean (*Coding*)

Desain harus diterjemahkan dalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

e. Pengujian (*Testing*)

Sesudah kode dibuat, pengujian program dimulai. Proses pengujian berfokus pada logika internal perangkat lunak, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional -- yaitu mengarahkan pengujian untuk menemukan kesalahan-kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil actual yang sesuai dengan hasil yang dibutuhkan.

f. Pemeliharaan (*Support*)

Perangkat lunak akan mengalami perubahan setelah disampaikan kepada pelanggan/ user (pengecualian yang mungkin adalah kepada perangkat lunak yang dilekatkan). Perubahan akan

terjadi karena kesalahan-kesalahan ditentukan, karena perangkat lunak harus disesuaikan untuk mengakomodasi perubahan-perubahan didalam lingkungan eksternalnya (contohnya perubahan yang dibutuhkan sebagai akibat perangkat peripheral atau sistem operasi yang baru), atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja. Pemeliharaan perangkat lunak mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi. Meskipun demikian, karena model ini melakukan pendekatan secara urut / sequential, maka ketika suatu tahap terhambat, tahap selanjutnya tidak dapat dikerjakan dengan baik dan itu menjadi salah satu kekurangan dari model ini. Selain itu, ada beberapa kekurangan pengaplikasian model ini, antara lain adalah sebagai berikut:

- a. Ketika problem muncul, maka proses berhenti, karena tidak dapat menuju ke tahapan selanjutnya. Bahkan jika kemungkinan problem tersebut muncul akibat kesalahan dari tahapan sebelumnya, maka proses harus membenahi tahapan sebelumnya agar problem ini tidak muncul. Hal-hal seperti ini yang dapat membuang waktu pengerjaan SE.
- b. Karena pendekatannya secara sequential, maka setiap tahap harus menunggu hasil dari tahap sebelumnya. Hal itu tentu membuang waktu yang cukup lama, artinya bagian lain tidak dapat mengerjakan hal lain selain hanya menunggu hasil dari tahap sebelumnya. Oleh karena itu, seringkali model ini berlangsung lama pengerjaannya.
- c. Pada setiap tahap proses tentunya dipekerjakan sesuai spesialisasinya masing-masing. Oleh karena itu, ketika tahap tersebut sudah tidak dikerjakan, maka sumber dayanya juga tidak terpakai lagi. Oleh karena itu, seringkali pada model proses ini dibutuhkan seseorang yang “multi-skilled”, sehingga minimal dapat membantu pengerjaan untuk tahapan berikutnya.

Masing-masing dari masalah tersebut bersifat riil. Tetapi paradigma siklus kehidupan klasik memiliki tempat yang terbatas namun penting di dalam kerja rekayasa perangkat lunak. Paradigma itu memberikan *template* di mana metode analisis, desain, pengkodean, pengujian, dan pemeliharaan bisa dilakukan. Siklus kehidupan klasik tetap menjadi model bagi rekayasa perangkat lunak yang paling luas dipakai. Walaupun memiliki kelemahan, secara signifikan dia lebih baik daripada pendekatan yang sifatnya asal kepada pengembang perangkat.

2.3 Tools untuk Pengembangan Perangkat Lunak

Microsoft Visual Basic 6.0

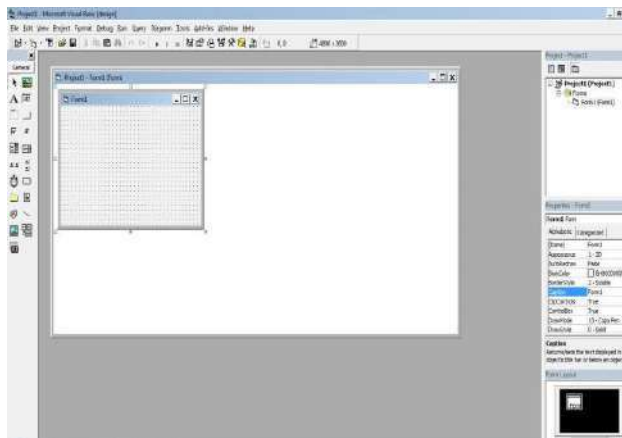
Microsoft Visual Basic (sering disingkat sebagai VB saja) merupakan sebuah bahasa pemrograman yang menawarkan Integrated Development Environment (IDE) visual untuk membuat program perangkat lunak berbasis sistem operasi Microsoft Windows dengan menggunakan model pemrograman. [2]

Visual Basic merupakan turunan bahasa pemrograman Basic dan menawarkan pengembangan perangkat lunak komputer berbasis grafik dengan cepat. Beberapa bahasa skrip seperti Visual Basic for Applications (VBA) dan Visual Basic Scripting Edition (VBScript), mirip seperti halnya Visual Basic, tetapi cara kerjanya yang berbeda.

Para programmer dapat membangun aplikasi dengan menggunakan komponen-komponen yang disediakan oleh Microsoft Visual Basic Program- program yang ditulis dengan Visual Basic juga dapat menggunakan Windows API, tapi membutuhkan deklarasi fungsi luar tambahan. Dalam pemrograman untuk bisnis, Visual Basic memiliki pangsa pasar yang sangat luas. Sebuah survey yang dilakukan pada tahun 2005 menunjukkan bahwa 62% pengembang perangkat lunak dilaporkan menggunakan berbagai bentuk Visual Basic, yang diikuti oleh C++, JavaScript, C#, dan Java.

Interface Antar Muka Visual Basic 6.0

Interface antar muka *Visual Basic 6.0* yang terdiri komponen penting dari lingkungan *Visual Basic* yang perlu Anda ketahui untuk lebih memudahkandalam memulai sebuah aplikasi berbasis *sual Basic 6.0* seperti terlihat pada gambar dibawah ini : [2]

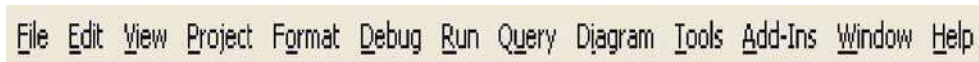


Gambar 2.13 Interface antar muka *Visual Basic 6.0*

Menu Bar

Menu Bar berfungsi untuk memilih tugas-tugas tertentu, seperti memulai, membuka dan

menyimpan project, mengompilasi project menjadi *file executable*(EXE), dan lain-lain.



Gambar 2.15 Tampilan Menu Bar

Main ToolBar

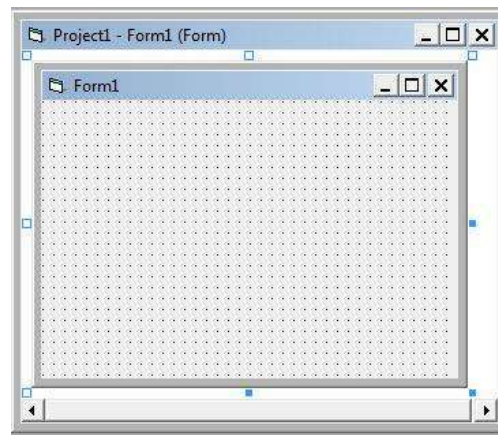
Main Toolbar memiliki fungsi yang sama seperti *menu bar*, tetapi berfungsi seperti jalanpintas karena lebih praktis dalam penggunaannya.



Gambar 2.16 Tampilan Main Toolbar

Form

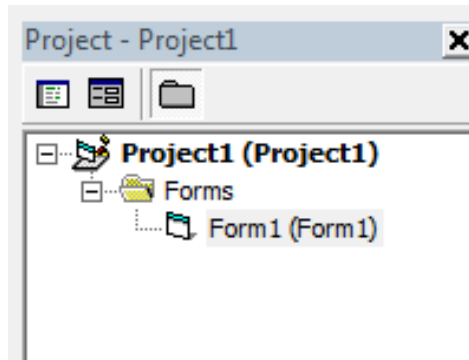
Form adalah lembar desain tampilan dari program yang dibuat. Form ini menjadipondasi tempat diletakkannya kontrol-kontrol yang dimiliki oleh *Visual Basic* sesuai dengan yang diinginkan.



Gambar 2.17 Tampilan Form

Jendela Project

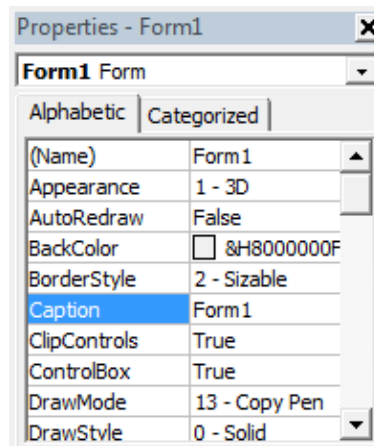
Project Windows (Jendela Proyek), menampilkan daftar form dan modul proyek. Proyek merupakan kumpulan dari *modul form*, *modul class*, *modul standart*, dan file sumber yang membentuk suatu aplikasi.



Gambar 2.18 Jendela Project

Jendela Properties

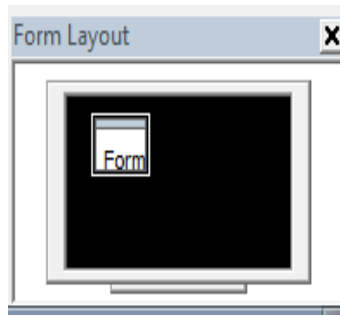
Properti Windows (Jendela Properti) berisi daftar struktur *setting property* yang digunakan pada sebuah objek yang terpilih. Kotak *drop-down* pada bagian atas jendela berisi daftar semua objek pada form yang aktif. Ada dua tab tampilan : *Alphabetic* (urut abjad) dan *Catagorized* (urut berdasarkan kelompok).



Gambar 2.19 Jendela Properti

Jendela Layout Form

Form Layout Windows (Jendela *Layout Form*) menampilkan posisi form relatif terhadap layar monitor pada saat program dijalankan atau dieksekusi (*Running/Compiling*).



Gambar 2.20 Jendela Layout Form

2.1.1 Keistimewaan Visual Basic 6.0

Visual Basic 6.0 juga memiliki beberapa keistimewaan antara lain :

- a. Menggunakan platform pembuatan program yang diberi nama dexplorer studio, yang memiliki tampilan dan saran yang sama dengan visual basic J ++. Dengan begitu anda dapat belajar bahasa pemograman lainnya dengan mudah, tepat dan cepat tanpa harus belajar lagi.
- b. Memiliki compiler yang handal yang dapat mendefenisikan *file Executable* efisien.
- c. Memiliki beberapa tambahan saran wizard yang baru. Wizard adalah sarana yang mempermudah didalam pembuatan aplikasi dengan mengotoritaskan tugas–tugas tertentu.
- d. Kemampuan membuat *active X* dan fasilitas internet yang lebih banyak.
- e. Sarana akses data yang lebih cepat dan handal untuk membuat database yang mempunyai kemampuan yang lebih tinggi.
- f. *Visual baic 6.0* memiliki variasi atau edisi yang disesuaikan dengan kebutuhan pemakainya.

Visual Basic menyediakan berbagai tipe data seperti pada tabel dibawahini antara lain :

Tipe–tipe data yang digunakan dalam *Visual Basic 6.0* antara lain :

Tabel 2.1 Tipe-tipe Data *Visual Basic*

Tipe	Ukuran Byte	karakter
Integer	2	%

Long	4	ε
Single	4	!
Double	8	#
Currency	8	@
String	1	\$
Byte	1	
Boolean	2	
Date	8	

Untuk melakukan berbagai manipulasi dan pengolahan data *Visual Basic* menyediakan berbagai operator antara lain :

a. Operator Aritmatika

Operator aritmatika digunakan untuk operasi aritmatika.

Tabel 2.2 Operator Aritmatika

Operator	Operasi
^	Pemangkatan
-	Tanda Negatif
*, /	Perkalian dan pembagian
\	Pembagian integer
Mod	Modulus sisa pembuangan
+, -	Penambahan dan pengurangan
+, ε	Penggabungan string

b. Operator Perbandingan

Operator perbandingan digunakan untuk membandingkan suatu data dengan data yang lain dan menghasilkan nilai logika benar atau salah, tetapi antara kedua data tersebut harus mempunyai nilai atau tipe data yang sama.

Tabel 2.3 Operator Perbandingan

Operator	Operasi
=	Sama dengan
<>	Tidak sama dengan
>	Lebih besar
<	Lebih kecil
< =	Lebih kecil atau sama dengan
> =	Lebih besar atau sama dengan

c. Operator Logika

Operator logika digunakan untuk mengekspresikan satu atau lebih daridua data (ekspresi) dan data logika (boolean) sehingga menghasilkan data logika yang baru.

Tabel 2.4 Operator Logika

Operator	Operasi
Not	Tidak
And	Dan
Or	Atau
Xor	Exclusive Or
Eqv	Ekivalen
Imp	Implikasi

Microsoft Office Access 2007

Microsoft Office Access merupakan produk dari *Microsoft* yang mempunyai fungsi dalam membangun *database* sistem. Dalam *Microsoft Office*, anda dapat melihat informasi dependensi antar objek *database*. Objek yang spesifik membantu mengelola *database over time* dan menghindari *error* yang berhubungan dengan kehilangan *record source*. Contohnya *query* permintaan triwulan pada *database* penjualan tidak dibutuhkan lagi, tetapi sebelum menghapusnya, anda harus mencari objek mana saja dalam *database* yang menggunakan *query* ini. Kemudian, anda dapat mengubah *record source* dari objek yang saling ketergantungan, atau menghapusnya, sebelum menghapus *query* permintaan triwulan. Menampilkan daftar lengkap objek yang saling ketergantungan membantu anda menghemat waktu dan permintaan *error*.

Keunggulan Microsoft Office Access

Microsoft Office Access memungkinkan untuk melindungi dari kode *Visual Basic for Application* (VBA) yang secara potensial tidak aman dengan mengatur level *Macro Security*. Anda dapat mengatur level keamanan sehingga setiap anda mengakses dan membuka *database* yang mengandung kode VBA, atau anda dapat secara otomatis melakukan blok *database* yang membentuk *source* yang tidak dikenal. Meskipun demikian, karena model ini melakukan pendekatan secara urut / sequential, maka ketika suatu tahap terhambat, tahap selanjutnya tidak dapat dikerjakan dengan baik dan itu menjadi salah satu kekurangan dari model ini. Selain itu, ada beberapa kekurangan pengaplikasian model ini, antara lain adalah sebagai berikut:

- d. Ketika problem muncul, maka proses berhenti, karena tidak dapat menuju ke tahapan selanjutnya. Bahkan jika kemungkinan problem tersebut muncul akibat kesalahan dari tahapan sebelumnya, maka proses harus membenahi tahapan sebelumnya agar problem ini tidak muncul. Hal-hal seperti ini yang dapat membuang waktu pengerjaan SE.
- e. Karena pendekatannya secara sequential, maka setiap tahap harus menunggu hasil dari tahap sebelumnya. Hal itu tentu membuang waktu yang cukup lama, artinya bagian lain tidak dapat mengerjakan hal lain selain hanya menunggu hasil dari tahap sebelumnya. Oleh karena itu, seringkali model ini berlangsung lama pengerjaannya.
- f. Pada setiap tahap proses tentunya dipekerjakan sesuai spesialisasinya masing-masing. Oleh karena itu, ketika tahap tersebut sudah tidak

dikerjakan, maka sumber dayanya juga tidak terpakai lagi. Oleh karena itu, seringkali pada model proses ini dibutuhkan seseorang yang “multi-skilled”, sehingga minimal dapat membantu pengerjaan untuk tahapan berikutnya.

Masing-masing dari masalah tersebut bersifat riil. Tetapi paradigma siklus kehidupan klasik memiliki tempat yang terbatas namun penting di dalam kerja rekayasa perangkat lunak. Paradigma itu memberikan *template* di mana metode analisis, desain, pengkodean, pengujian, dan pemeliharaan bisa dilakukan. Siklus kehidupan klasik tetap menjadi model bagi rekayasa perangkat lunak yang paling luas dipakai. Skalipun memiliki kelemahan, secara signifikan dia lebih baik daripada pendekatan yang sifatnya asal kepada pengembang perangkat.

3.1.1 Sejarah Singkat House Of Leather

House Of Leather adalah sebuah perusahaan kerajinan tangan yang berdomisili di Bandung. Setelah memasarkan produk lewat cara konsinyasi, akhirnya pada sekitar tahun 2000-an, House Of Leather resmi berdiri dengan membuka workshop sendiri di Jl. Cikutra No. 18/148 B.

Home Industry ini berdiri atas mitra binaan Pertamina dengan peminjaman modal awal. Selain mendapatkan modal, Pertamina juga kerap mengajak perusahaan ikut serta dalam berbagai pameran. Keikutsertaan dalam berbagai pameran tersebut menjadi langkah awal *House of Leather* dikenal oleh masyarakat luas.

Bahan-bahan yang digunakan dalam pembuatan tas ini antara lain kulit sapi, buaya, dan ular. Mayoritas bahan kulit binatang tersebut didatangkan dari sebuah pabrik di Tangerang, Karawang, Surabaya. Dalam kurun waktu satu minggu produk dapat habis terjual.

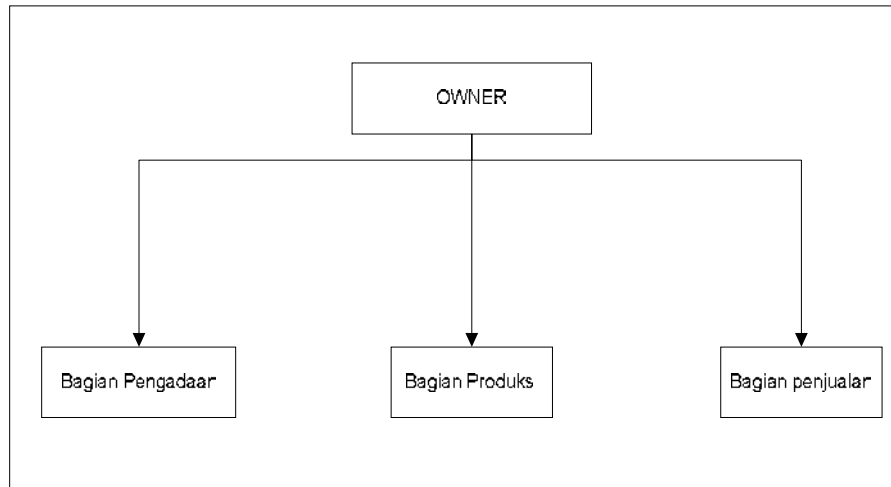
Kedepannya, House Of Leather mencoba memperluas pasar dengan memproduksi aksesoris lainnya yang akan melengkapi fashion konsumen dengan koleksi ikat pinggang (belt), dompet, sandal dan sepatu, juga gift item, seperti gantungan kunci dan lainnya.

Sejauh ini, pihak *owner* sangat puas dengan apa yang telah diraih, namun mereka masih akan terus mengembangkan usaha dalam bidang penjualan Tas ini

, dan akan selalu berusaha untuk memberikan pelayanan terbaik terhadap

konsumen agar kedepannya dapat menjadi perusahaan penjualan Tas yang terkemuka.

3.1.2 Struktur Organisasi



Gambar 3.1 Struktur Organisasi

3.1.3 Uraian Tugas

a. Owner

1. Mengendalikan perusahaan pada umumnya
2. Mendorong karyawan agar bekerja sesuai dengan sasaran yang direncanakan.
3. Menjaga serta menjamin kelangsungan perkembangan perusahaan.
4. Menentukan kebijakan atau pengambilan keputusan segala masalah operasional perusahaan.

b. Bagian Pengadaan

1. Merancang hubungan yang tepat dengan supplier.
2. Klasifikasi barang yang harus dibeli oleh perusahaan seperti bahan baku dan komponen untuk kebutuhan produksi

c. Bagian Produksi

Tugas utama dari bagian produksi dalam kaitannya dengan pencapaian tujuan perusahaan secara umum adalah berusaha mencapai biaya produksi yang rendah, mutu produk yang tinggi, tanggapan yang cepat atas

permintaan, dan fleksibilitas untuk membuat beragam barang yang sesuai dengan selera dan spesifikasi pelanggan.

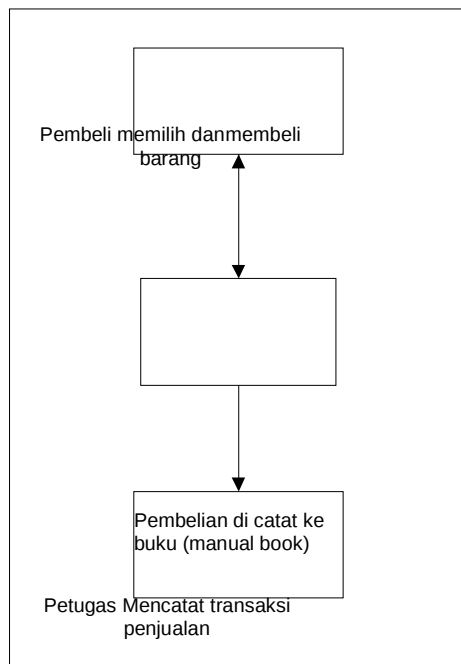
d. Bagian Penjualan

1. Melakukan transaksi penjualan.
2. Membuat nota transaksi penjualan.
3. Membuat laporan stok barang untuk pimpinan

3.2 Deskripsi Sistem Lama

3.2.1 Analisis sistem yang sedang berjalan

Berdasarkan hasil observasi dan wawancara langsung kepada owner di house of leather sistem yang berjalan saat ini masih di tulis manual menggunakan buku (manual book) dari proses pencatatan data supplier, data barang, stok barang, data penjualan barang, laporan penjualan, dll. Sistem ini di rasa masih kurang optimal karena masih menggunakan buku sebagai media pencatatannya.



Gambar 3.2. Gambaran Sistem lama aplikasi penjualan

3.2.2 Skenario Proses Informasi Aplikasi penjualan

Berikut ini alur proses sistem lama seperti di gambarkan di atas :

1. Pembeli datang langsung ke workshop dan memilih barang yang akan di beli kemudian di serahkan langsung ke bagian penjualan
2. Bagian penjualan membuat bon pembayar yang terdiri dari dua rangkap dan masing-masing akan di berikan kepada pembeli dan yang satu lagi di simpan untuk arsip
3. Bon penjualan akan di catat ke buku (manual book) untuk laporan.

3.2.3 Kelemahan sistem lama :

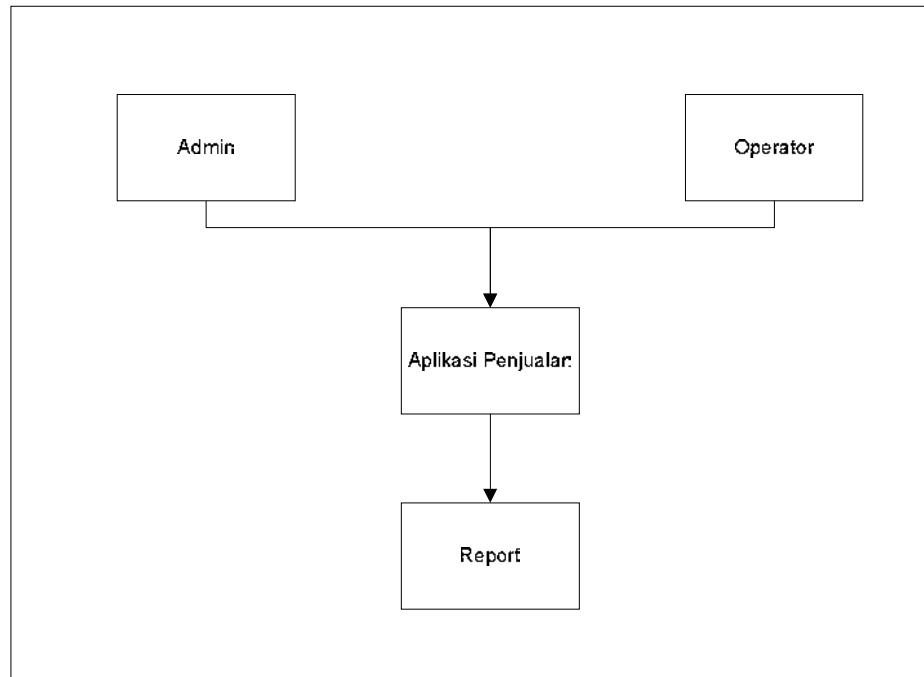
1. Sistem penjualan tas yang belum terkomputerisasi menyulitkan penjual dalam melakukan transaksi, setiap transaksi akan dicatat secara manual melalui media buku begitu pula dengan penghitungan pendapatan maupun pengeluaran yang masih dilakukan secara manual dengan menggunakan kalkulator misalnya.
2. Kemungkinan data hilang sangat besar karena buku / kertas untuk mencatat mudah hilang atau tercecer.
3. Dalam pembuatan laporan pun terhambat / kurang efisien karena harus diolah manual datanya.

Dari hasil analis sistem lama maka di butuhkan suatu aplikasi terkomputerisasi yang dapat memudahkan dalam penyimpanan data dan pengolahan data penjualan, sekaligus memudahkan dalam pembuatan laporan sesuai dengan kebutuhan.

3.3 Deskripsi Sistem Baru

Fungsi utama perancangan aplikasi penjualan di UKM ini antara lain :

1. Memudahkan dalam pengolahan data dan penyimpanan data (datanya aman karena tidak tercecer).
2. Memudahkan dalam pencatatan data supplier, data barang, stok barang, data penjualan barang, laporan penjualan, sampai dengan proses pembuatan report dapat dilakukan dengan mudah karena datanya tersedia dan tersimpan menyatu di dalam aplikasi penjualan ini.



Gambar 3.3 Gambaran Sistem Baru Aplikasi Penjualan

3.4 Skenario Proses Aplikasi penjualan

1. Admin dan operator melakukan login terlebih dahulu
2. Sistem menampilkan menu utama aplikasi penjualan yang terdiri data supplier, data barang, stok barang, data penjualan barang.
3. Menampilkan laporan yang diperlukan

3.5 Analisis Kebutuhan Perangkat Lunak yang di Kembangkan dengan Model Use Case Business

Analisis di deskripsikan dengan menggunakan *Use Case Business Diagram*, *Scenario Use Case Business* dan *Activity diagram*.

3.5.1 Definisi Aktor

Deskripsi aktor merupakan penjelasan dari apa yang dilakukan oleh aktor- aktor yang terlibat dalam perangkat lunak yang akan dibangun. Adapun deskripsi dari aktor-aktor yang terlibat dalam Aplikasi penjualan adalah sebagai berikut :

Tabel. 3.1 Deskripsi Aktor

No	Aktor	Deskripsi
1.	Admin	Melakukan tugas login dan berhak atas semua

		akses pada aplikasi penjualan
2.	<i>Operator</i>	Melakukan tugas login dan hak aksesnya hanya pada data penjualan, laporan penjualan, backup dan restore

3.5.1 Deskripsi Use Case

Deskripsi *Use Case* merupakan penjelasan dari *use case* atau proses-proses yang berlangsung di dalam kegiatan Sistem. Adapun deskripsi dari *Use Case* Sistem Aplikasi penjualan di UKM adalah sebagai berikut :

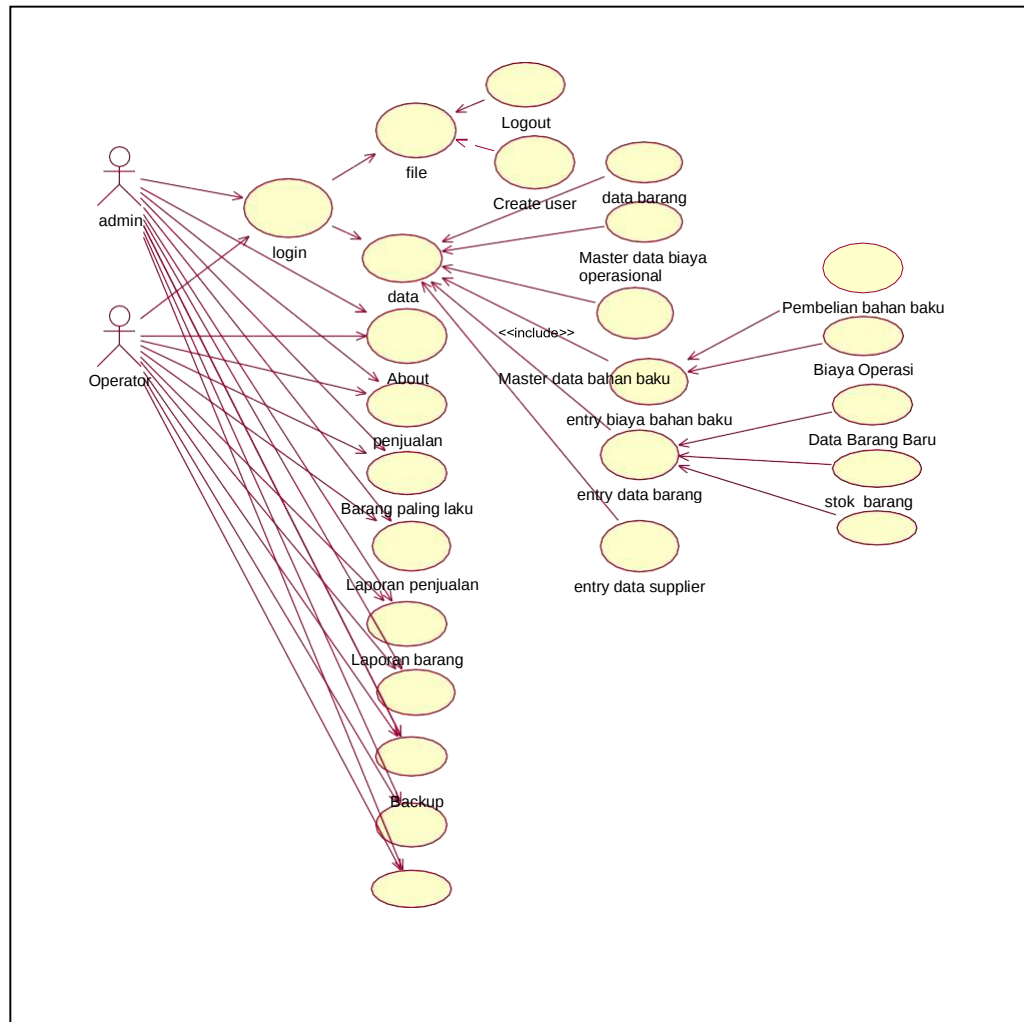
Tabel 3.2 Deskripsi Use Case

No	Usecase	Deskripsi
1.	Login	Tahap awal sebelum admin dan operator masuk ke sistem utama untuk melakukan pengolahan
2.	Data Barang	Mengetahui data barang apa saja yang Di jual
3.	Master Data Biaya Operasional	Master data untuk pengeluaran bulanan
4.	Master Data Bahan Baku	Master data bahan baku yang di gunakan untuk pembuatan tas
5.	PembelianBaku bahan	Proses pengarsipan transaksi pembelian bahan baku ke supplier
6.	Biaya Operasi	Pencatatan pengeluaran biaya operasional dan lain-lain
7.	Data barang baru	Proses menambah dan mengetahui harga barang baru yang sudah di tambahkan keuntungan
8.	Stok barang	Mengetahui stok barang awal dan akhir, Proses mengubah dan menambahkan stok barang
9.	Update diskon	Proses pengupdatean data diskon sesuai dengan jumlah barang yang di beli
10.	Entry data Supplier	Pengarsipan data-data supplier
11.	penjualan	Proses transaksi penjualan
12.	Barang paling laku	Dapat mengetahui penjualan baang paling laku setiap bulan atau setiap tahunnya
13.	Laporan penjualan	Admin dan operator dapat mengetahui dan Mencetak laporan penjualan
14.	Laporan Barang	Memeriksa laporan barang yang sudah terjual

15.	Backup Database	Melakukan penyimpanan database
16.	Restore Database	Melakukan pengambilan database yang telah di backup
17.	View Data	Dapat Menampilkan Data Supplier,data barang,stok barang dan data penjualan

3.5.2 Use Case Aplikasi Penjualan di UKM

Diagram *Use Case* Aplikasi penjualan di UKM adalah sebagai berikut :



Gambar 3.4 *Use Case Business Diagram* Aplikasi penjualan di UKM

3.5.3 Skenario Use Case Aplikasi Penjualan

Skenario (flow of event) untuk masing-masing use case dari aplikasipenjualan di UKM adalah sebagai berikut :

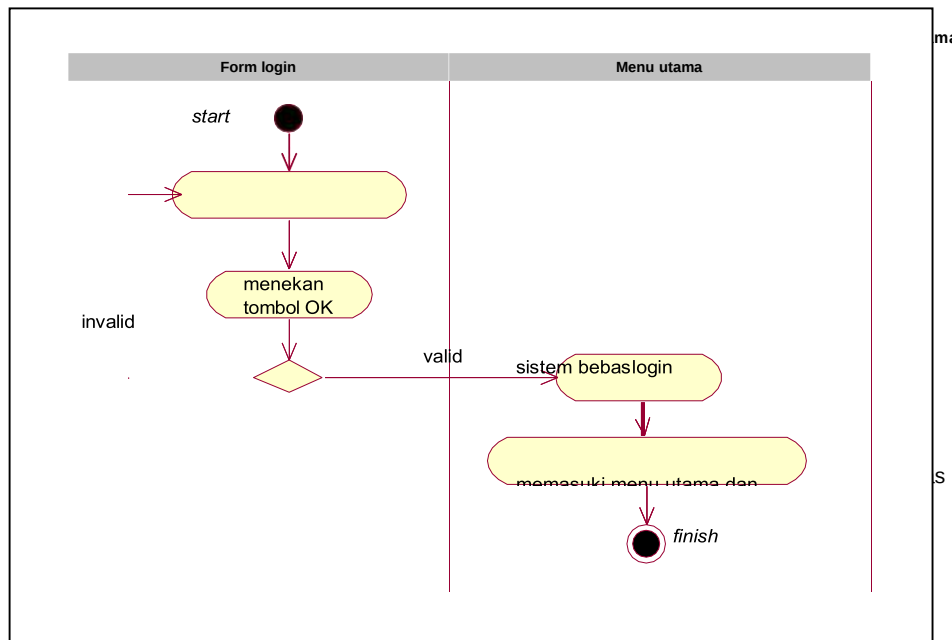
1. Skenario Use Case Login Admin

Tabel 3.3 Skrenario Use Case Login Admin

Name	Login <i>Admin</i>
Level	Pengguna
Aktor	<i>Admin</i>
Goal	Untuk mengelola semua data, serta untuk memasuki halaman menu utama
Precondition	<i>Admin</i> belum valid dan belum berhasil masuk kehalaman menu utama
Postcondition	<i>Admin</i> telah <i>valid</i> untuk masuk halaman menu utama
Steps	. <i>Admin</i> memasukkan <i>ID user</i> dan <i>password</i> . <i>Admin</i> menekan tombol <i>ok</i> <i>Admin</i> telah berhasil login memasuki ke halaman menu utama dan berhak atas semuaakses pada aplikasi penjualan

2. Activity Diagram Use Case Login Admin

Activity Diagram adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu Use Case. Activity Diagram berikut menggambarkan proses login admin terhadap isi aplikasi penjualan sebagai berikut:



Gambar 3.5 Activity Diagram Use Case Login Admin

3. Skenario Use Case Login Operator

Skenario (flow of event) untuk masing-masing use case dari aplikasi penjualan diUKM adalah sebagai berikut :

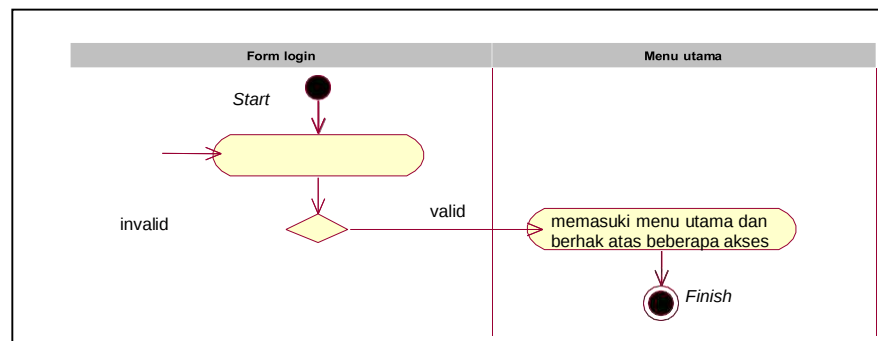
Tabel 3.4 Skrenario Use Case Login operator

Name	Login operator
Level	Pengguna
Aktor	operator
Goal	Untuk mengelola sebagian data, serta untuk memasuki halaman menu utama
Precondition	operator belum valid dan belum berhasil masuk kehalaman menu utama
Postcondition	operator telah valid untuk masuk halaman menu utama
Steps	. operator memasukkan ID user dan password

	operator telah berhasil login memasuki ke halaman menu utama dan berhak atas beberapa akses seperti penjualan, barang paling laku, laporan penjualan, laporan barang, backup, restore, view data pada aplikasi penjualan
--	--

4. Activity Diagram Use Case Login Operator

Activity Diagram adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu *Use Case*. *Activity Diagram* berikut menggambarkan proses login operator terhadap isi aplikasi penjualan sebagai berikut:



Gambar 3.6 Activity Diagram Use Case Login Operator

PERANCANGAN

4.1 Perancangan Sistem

Pada House of leather akan dibuat sebuah perancangan aplikasi penjualan untuk UKM. Perancangan aplikasi ini di bangun bertujuan untuk memudahkan dalam mengolah data, mencatat data dan penjualan.

Perancangan aplikasi penjualan untuk UKM yang di bangun ini bersifat *object oriented* (berorientasi objek) dengan menggunakan *Unified Modeling Language (UML)* sebagai bahasa pemodelan. Pembangunan Aplikasi penjualan ini dilakukan dengan menggunakan *tools* utama sebagai berikut :

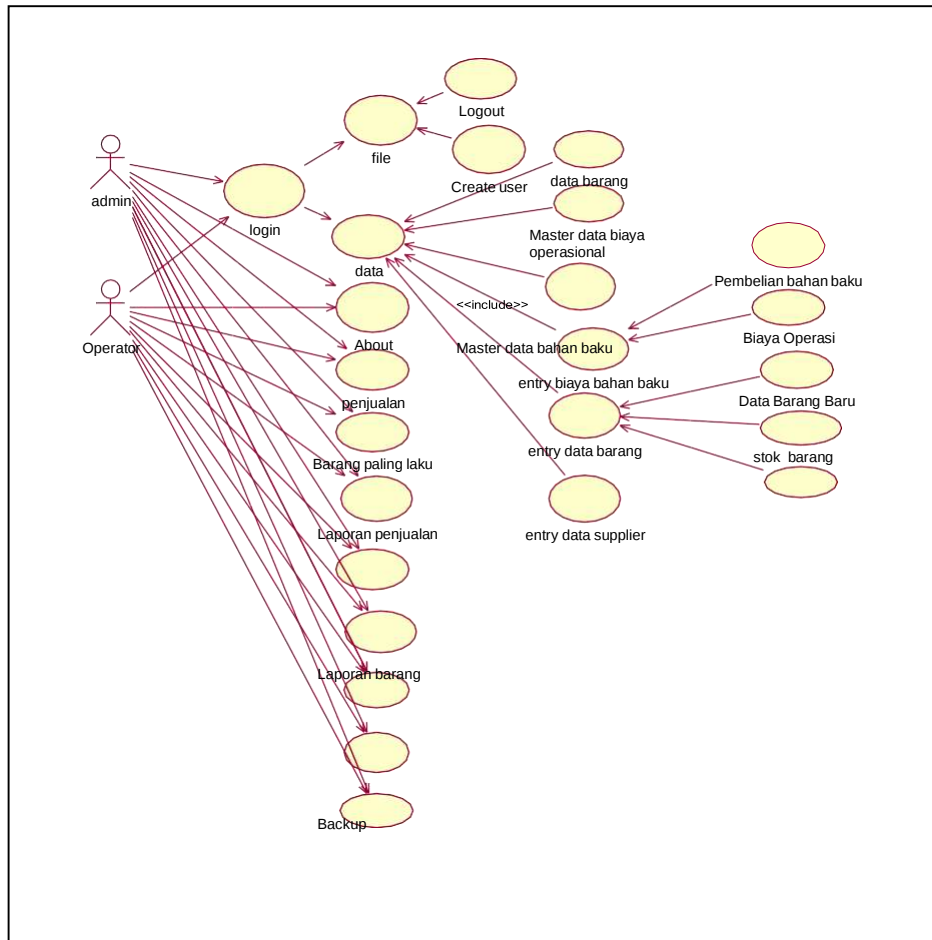
1. *Visual basic 6.0* merupakan sebuah bahasa pemrograman untuk pembuatan aplikasi.
2. *Microsoft office Access* sebagai *Database*.
3. *Cristal Report* untuk mencetak hasil laporan

4.2 Model Use Case

Model *use case* menjelaskan mengenai aktor-aktor yang terlibat dengan perangkat lunak yang dibangun beserta proses-proses yang ada didalamnya.

4.2.1 Use Case Diagram

Diagram use case dari Perancangan Aplikasi penjualan di UKM adalah sebagai berikut :



Gambar 4.1 Use Case Business Diagram Aplikasi Penjualan di UKM

4.2.2 Definisi Actor

Definisi aktor merupakan penjelasan dari apa yang dilakukan oleh aktor-aktor yang terlibat dalam perangkat lunak yang dibangun. Adapun deskripsi dari aktor-aktor yang terlibat dalam perancangan Aplikasi penjualan di UKM sebagai berikut :

Tabel 4.1 Definisi Aktor

No.	Aktor	Deskripsi
1.	Admin	Melakukan tugas <i>login</i> dan memanipulasi (<i>Insert, Update, dan Delete</i>) semua isi content menu aplikasi penjualan.
2.	Operator	Melakukan login, mengolah penjualan barang, laporan, backup dan restore database.

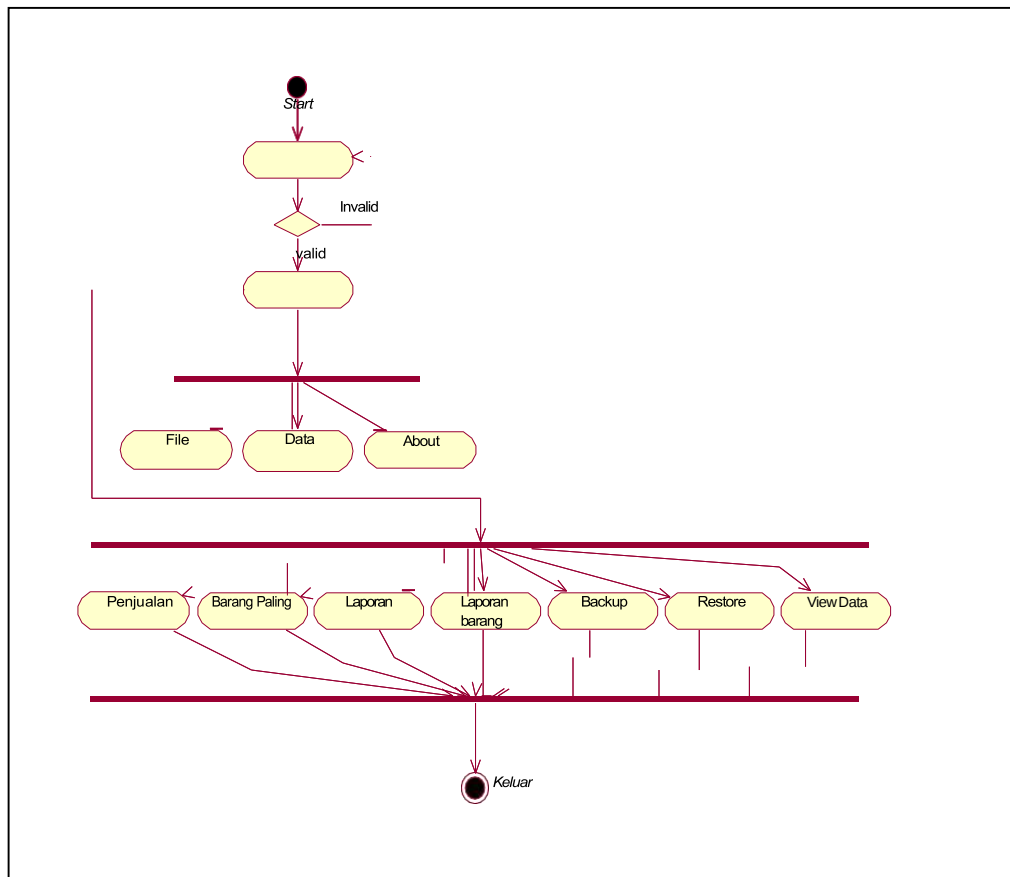
4.2.3 Deskripsi Use Case**Tabel 4.2 Deskripsi Use Case**

No.	Use Case	Deskripsi
1.	Login	Tahap awal sebelum admin dan operator masuk ke sistem utama untuk melakukan pengolahan
2.	Data Barang	Mengetahui data barang apa saja yang Di jual
3.	Master Data Biaya Operasional	Master data untuk pengeluaran bulanan
4.	Master Data Bahan Baku	Master data bahan baku yang di gunakan untuk pembuatan tas
5.	Pembelian bahan Baku	Proses pengarsipan transaksi pembelian bahan baku ke supplier
6.	Biaya Operasi	Pencatatan pengeluaran biaya operasional dan lain-lain
7.	Data barang baru	Proses menambah dan mengetahui harga barang baru yang sudah di tambahkan keuntungan
8.	Stok barang	Mengetahui stok barang awal dan akhir, Proses mengubah dan menambahkan stok barang
9.	Update diskon	Proses pengupdatean data diskon sesuai dengan jumlah barang yang di beli
10.	Entry data Supplier	Pengarsipan data-data supplier
11.	penjualan	Proses transaksi penjualan
12.	Barang paling laku	Dapat mengetahui penjualan barang paling laku setiap bulan atau setiap tahunnya
13.	Laporan penjualan	Admin dan operator dapat mengetahui dan Mencetak laporan penjualan

14.	Laporan Barang	Memeriksa laporan barang yang sudah terjual
15.	Backup Database	Melakukan penyimpanan database
16.	Restore Database	Melakukan pengambilan database yang telah di backup
17.	View Data	Dapat Menampilkan Data Supplier,data barang,stok barang dan data penjualan

4.3 Activity Diagram

Activity diagram adalah salah satu cara untuk memodelkan event-event yang terjadi dalam suatu *use case*. Berikut ini activity diagram dari Perancangan Aplikasi Penjualan di UKM :



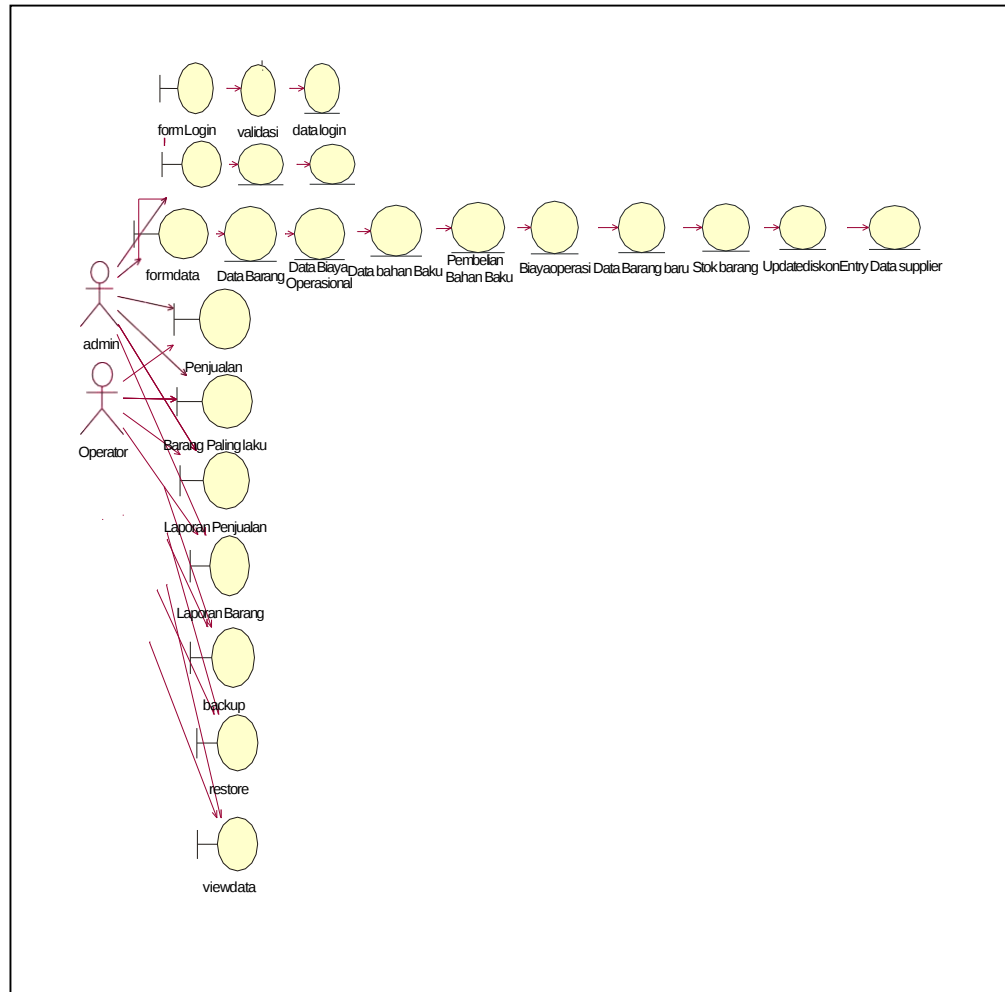
Gambar 4.2 Activity Diagram

4.4 Realisasi Use Case Tahap Analisis

Realisasi *use case* tahap analisis terdiri dari *class analisis* Diagram yang menggambarkan interaksi setiap objek dari kelas analisis yang terlibat di dalam *use case* tersebut.

4.4.1 Class Analysis Digram

Class analysis Diagram merupakan penggambaran keterkaitan objek- objek yang ada dalam dalam lingkungan sistem informasi yang dikembangkan. Berikut adalah *class analysis* Diagram dari perancangan Aplikasi Penjualan di UKM :



Gambar 4.3 Class Analisis Diagram

4.4.2 Class Analysis

Tabel 4.3 Class Analysis

No	Nama Kelas	Jenis
1.	Form login	<i>Boundary</i>
2.	Validasi	<i>Control</i>
3.	Form file	<i>Boundary</i>
4.	Create user	<i>Entity</i>
5.	Form data	<i>Boundary</i>
6.	Data Barang	<i>Entity</i>
7.	Data biaya Operasional	<i>Entity</i>
8.	Data Bahan Baku	<i>Entity</i>
9.	Pembelian Bahan baku	<i>Entity</i>
10.	Biaya Operasi	<i>Entity</i>
11.	Data Barang Baru	<i>Entity</i>
12.	Stock Barang	<i>Entity</i>
13.	Update diskon	<i>Entity</i>
14.	Entry data supplier	<i>Entity</i>
15.	Penjualan	<i>Boundary</i>
16.	Barang Paling Laku	<i>Boundary</i>
17.	Laporan penjualan	<i>Boundary</i>
18.	Laporan barang	<i>Boundary</i>
19.	Backup Database	<i>Boundary</i>
20.	Restore database	<i>Boundary</i>
21.	View Data	<i>Boundary</i>

4.4.3 Tanggung Jawab dan Atribut

Tanggung jawab merupakan proses-proses yang dilakukan pada kelas kontrol, sedangkan atribut merupakan keterangan dari setiap *boundary*, *entity* dan *control*. Daftar tanggung jawab dan atribut dari perancangan Aplikasi penjualan di UKM adalah sebagai berikut :

Tabel 4.4 Tanggung Jawab dan Atribut

Nama Kelas	Daftar Tanggung jawab	Daftar Atribut
FormLogin	<i>Login , logout</i>	1.Username 2.Password
Create User	<i>getCreate user</i>	1.id user 2.username 3.Password 4. ketik ulang password 5. posisi
Data barang	<i>getData barang</i>	1. id barang 2. nama barang
Data Biaya operasional	<i>getData biaya oprasional</i>	1. id operasional 2. nama operasional
Data Bahan Baku	<i>getData Bahan Baku</i>	1.id bahan baku 2.id supplier 3.nama supplier 4.keterangan barang
Pembelian bahan baku	<i>getEntry biaya produksi</i>	1.tanggal pembelian 2.id bahan baku 3.nama supplier 4.keterangan bahan 5.harga / barang 6.id barang 7.nama barang 8.jumlah bahan baku 9.total harga
Biaya Operasi	<i>getEntry biaya produksi</i>	1.bulan operasional 2.id operasional

		3. keterangan operasional 4. total biaya operasional
Data Barang Baru	<i>getEntry</i> data barang	1.biaya bulan 2.id barang 3.nama barang 4.stok barang 5.keuntungan per barang 6.harga per barang
Stok barang	<i>getEntry</i> data barang	1.id barang 2.nama barang 3.tambah stok
Update diskon	<i>getEntry</i> data barang	1. barang diatas 2. mendapatkan diskon 3.tanggal update
Entry data Supplier	<i>getEntry</i> data supplier	1.no supplier 2.nama supplier 3.kota 4. telepon 5. keterangan 6.harga per barang
Penjualan	<i>get</i> penjualan	1.no faktur 2.id barang 3.nama barang 4.harga / barang 5.jumlah barang 6.harga barang 7.total harga
Barang paling laku	<i>get</i> barang paling laku	Pilih berdasarkan
Laporan penjualan	<i>get</i> Laporan penjualan	Pilih berdasarkan
Laporan Barang	<i>get</i> Laporan barang	1.pilih berdasarkan 2,kata kunci

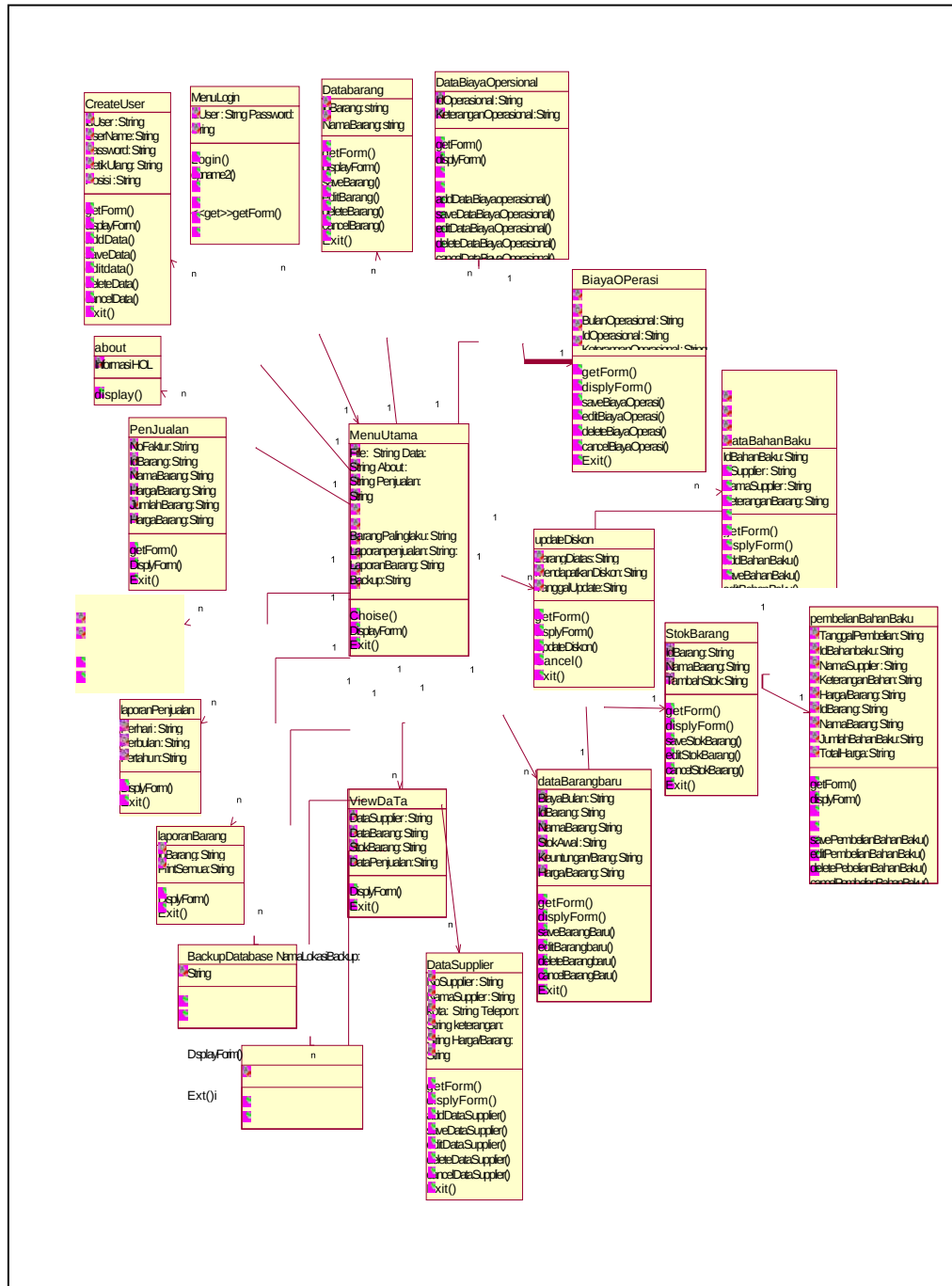
Backup database	<i>getbackup database</i>	Nama lokasi backup
Restore database	<i>getRestore database</i>	Nama lokasi restore
View data	<i>getView Data</i>	Plih berdasarkan

4.5 Realisasi Use Case Tahap Perancangan

Realisasi *use case* tahap perancangan digambarkan dalam *class* Diagram dan *sequence* Diagram, *collabaration* yang menggambarkan interaksi setiap objek dari kelas perancangan yang terlibat di dalam *use case* tersebut.

4.5.1 Class Diagram

Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas yang ada dalam sistem perangkat lunak yang akan dikembangkan. *Class* Diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan. Berikut ini digambarkan *class* diagram dari perancangan Aplikasi penjualan di UKM :



Gambar 4.4 Class Diagram

4.5.2 Kamus Data

Berikut ini akan dijelaskan tabel – tabel yang digunakan dalam perancanganproses *Unified Modelling Language* (UML).

1. Tabel Admin

Nama Tabel : TBAdminPrimary Key : ID_User

Keterangan : Tabel ini berisikan data – data admin dalam house of leather

Tabel 4.5 Tabel Admin

Field Name	Data Type	Field Size	Description
ID_User	Text	5	Primary key
UserName	Text	10	Nama user
Password1	Text	5	Password user
Level1	Text	8	Level user

2. Tabel Data Produk

Nama Tabel : Data ProdukPrimary Key :

ID_Produk

Keterangan : Tabel ini berisikan data nama-nama produk yang ada di houseof leather

Tabel 4.6 Tabel Data Produk

Field Name	Data Type	Field Size	Description
ID_Produk	Text	8	Primary Key
Nama	Text	15	Nama Produk

3. Tabel Data Master Biaya Produksi

Nama Tabel : DataMasBiayaOp

Primary Key : ID_BiayaOp

Keterangan : Tabel ini berisikan data biaya operasi yang ada di house ofleather

Tabel 4.7 Tabel Data Master biaya Operasi

Field Name	Data Type	Field Size	Description
ID_BiayaOP	Text	5	Primary Key
Keterangan	Text	20	Keterangan

4. Tabel Data Master Bahan Baku

Nama Tabel : DataMasBaku

Primary Key : ID_BahanBaku

Foreign Key : ID_Supplier

Keterangan : Tabel ini berisikan data bahan baku berserta nama supplieryang ada di house of leather

Tabel 4.8 Tabel Master Bahan Baku

Field Name	Data Type	Field Size	Description
ID_BahanBaku	Text	8	Primarykey
ID_Supplier	Text	8	Foreign key
Nama	Text	10	Nama Supplier
Keterangan	Text	10	Keterangan

5. Tabel Data Bahan Baku

Nama Tabel : DataBhnBaku

Primary Key : ID_Bahan

Foreign Key : ID_Produk

Keterangan : Tabel ini berisikan data nama bahan yang ada di house ofleather

Tabel 4.9 Tabel Data Bahan Baku

Field Name	Data Type	Field Size	Description
ID_Bahan	Text	8	Primary Key
ID_Produk	Text	8	Foreign Key
Nama_Barang	Text	20	Nama
Jumlah	Number	3 Digit	Jumlah
Jenis_Barang	Text	20	Jenis
Harga	Currency	-	Harga bahan baku
Tanggal	Date / Time	-	Tanggal pembelian bahan baku

6. Tabel Data Biaya

Nama Tabel : DataBiayaPrimary Key :

ID_Biaya

Keterangan : Tabel ini berisikan data biaya-biaya pengeluaran yang ada dihouse of leather

Tabel 4.10 Tabel Data Biaya

Field Name	Data Type	Field Size	Description
ID_Biaya	Text	8	Primary Key
Keterangan	Text	20	keterangan
Ttl_biaya	Currency	-	Total biaya
tanggal	Date / Time	-	tanggal

7. Tabel Data Barang

Nama Tabel : DataBarangPrimary Key :

ID_Produk

Keterangan : Tabel ini berisikan data barang yang ada di house of leather

Tabel 4.11 Tabel Data Barang

Field Name	Data Type	Field Size	Description
ID_Produk	Text	8	Primarykey
Stok_Awal	Number	2 Digit	Stok barang
Keuntungan_PerBarang	Currency	-	Keuntungan perbarang
Harga	Currency	-	Harga barang
Tanggal	Date / Time	-	tanggal

8. Tabel Data Stok

Nama Tabel : DataStok Primary Key :

ID_Produk

Keterangan : Tabel ini berisikan stok barang yang ada di house of leather

Tabel 4.12 Tabel Data Stok

Field Name	Data Type	Field Size	Description
ID_Produk	Text	8	Primary key
Stok_Awal	Number	2 Digit	Stok barang
Sisa_Stok	Number	2 Digit	Sisa stok
Tanggal	Date / Time	-	tanggal

9. Tabel Data Diskon

Nama Tabel : DataDiskonPrimary Key : ID_Dis

Keterangan : Tabel ini berisikan diskon yang akan di update di house ofleather

Tabel 4.13 Tabel Data Diskon

Field Name	Data Type	Field Size	Description
ID_Dis	Text	8	Primary key
Jumlah_Barang	Number	3 Digit	Jumlah barang
Persen	Number	3 Digit	Diskon
tanggal	Date /Time	-	tanggal

10. Tabel DataSupplier

Nama Tabel : DataSupplierPrimary Key : ID_Supl

Keterangan : Tabel ini berisikan data supplier di house of leather

Tabel 4.14 Tabel Data Supplier

Field Name	Data Type	Field Size	Description
ID_Supl	Text	8	Primary key
Nama	Text	10	Nama supplier
Kota	Text	10	Kota
Telepon	Text	15	Telepon
Keterangan	Text	255	keterangan
Harga	Curency	-	Harga Bahan Baku
Jenis	Text	10	Jenis

11. Tabel DataPenjualan

Nama Tabel : DataPenjualanPrimary Key :

ID_Faktur

Keterangan : Tabel ini berisikan data penjualan di house of leather

Tabel 4.15 Tabel Data Penjualan

Field Name	Data Type	Field Size	Description
ID_Faktur	Text	8	Primary Key
Total_item	Number	3 Digit	Total item
Total_Harga	Currency	-	Total harga
Persen	Number	3 Digit	Jumlah diskon
Total_Bayar	Currency	-	Total bayar
Total_Cash	Currency	-	Total cash
Kembalian	Currency	-	Kembalian
Tanggal	Date /Time	-	Tanggal
NamaUser	Text	10	Nama user

12. Tabel Detail Penjualan

Nama Tabel : DetailpenjualanPrimary Key :

ID_Faktur Foreign Key : ID_Produk

Keterangan : Tabel ini berisikan data detail penjualan di house of leather

Tabel 4.16 Tabel Detail penjualan

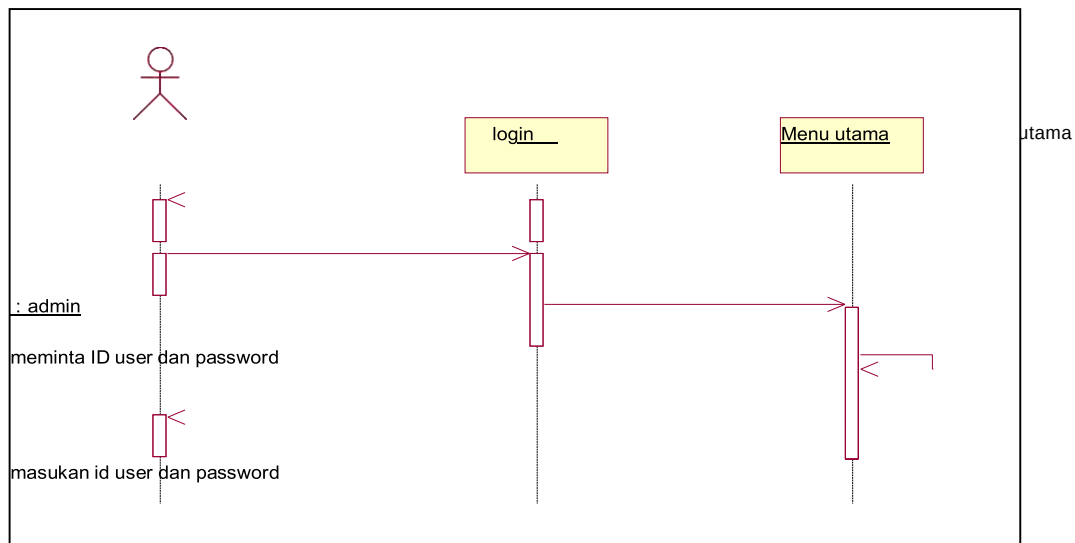
Field Name	Data Type	Field Size	Description
ID_Faktur	Text	8	Primary Key
ID_Produk	Text	8	Foreign Key
Jumlah_Barang	Number	3 Digit	Jumlah barang
Jumlah_harga	Currency	-	Jumlh harga

4.6 Struktur Organisasi Obyek dan Pesan

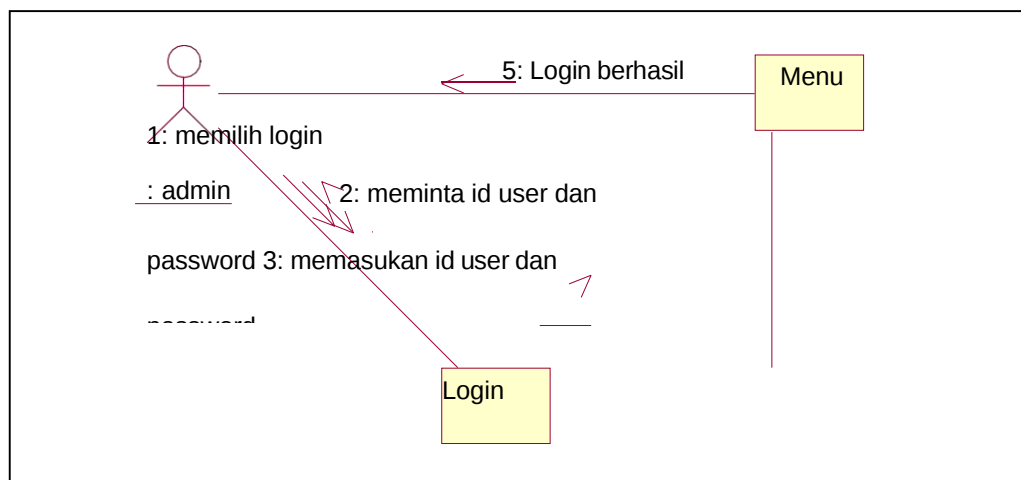
Sub bab ini menggambarkan aspek keterurutan waktu dari pesan yang disampaikan dan juga menggambarkan aspek struktur organisasi objek yang mengirim dan menerima pesan. Penggambaran ini dimodelkan dalam *Sequence Diagram* dan *Collaboration Diagram* beserta uraian tekstual

4.6.1 Scenario Sequence Diagram dan Collaboration Diagram login

Name	Login
Aktor	(user) Admin dan Operator
Goal	Sistem mengijinkan user ke menu selanjutnya
Precondition	Sistem aktif
Postcondition	Login berhasil
Steps	1. User menyalakan komputer dan membuka aplikasi penjualan 2. User memilih Login dan kemudian memasukkan id user serta password



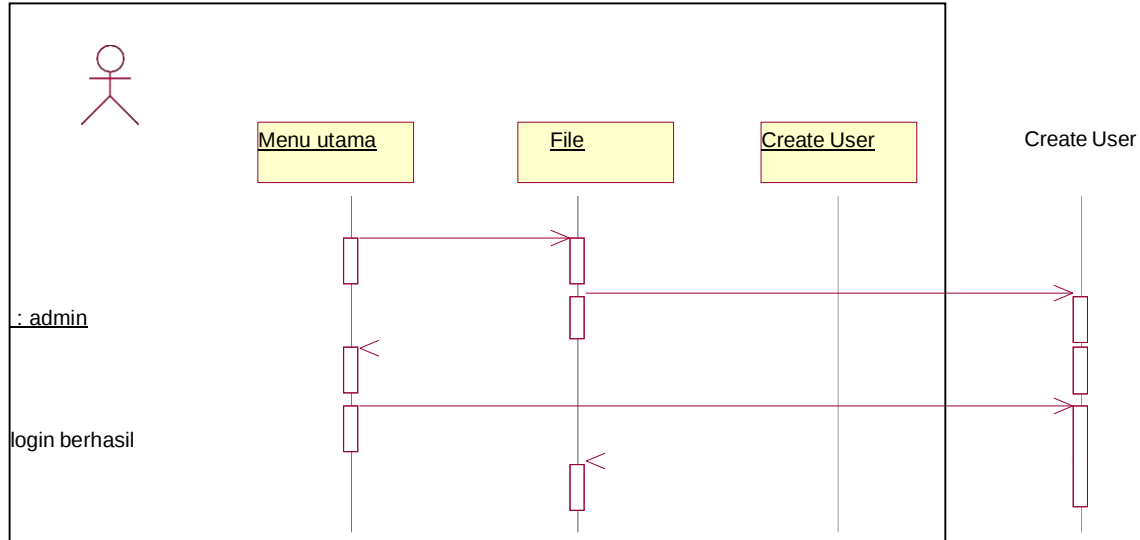
Gambar 4.5 Sequence Diagram Login



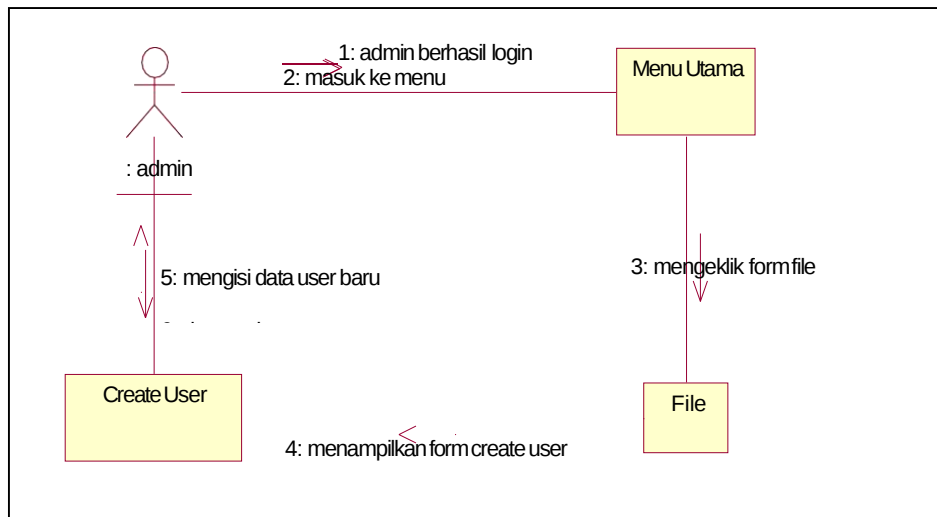
Gambar 4.6 Collaboration Diagram Login

4.6.2 Scenario Sequence Diagram dan Collaboration Diagram Create user

Name	Create user
Aktor	User (Admin)
Goal	Input data user, ubah data user dan hapusdata user berhasil dilakukan.
Precondition	Membuka menu Entry data user
Postcondition	Berhasil memasukkan user baru, mengubah serta menghapus user
Step	1. Admin membuka menu data dan milih create user 2. Admin memasukkan data user baru. 3. Admin menyimpan data baru yang telah dimasukkan dengan mengklik simpan, selain itu admin juga dapat melakukan pengubahandata serta penghapusan data.



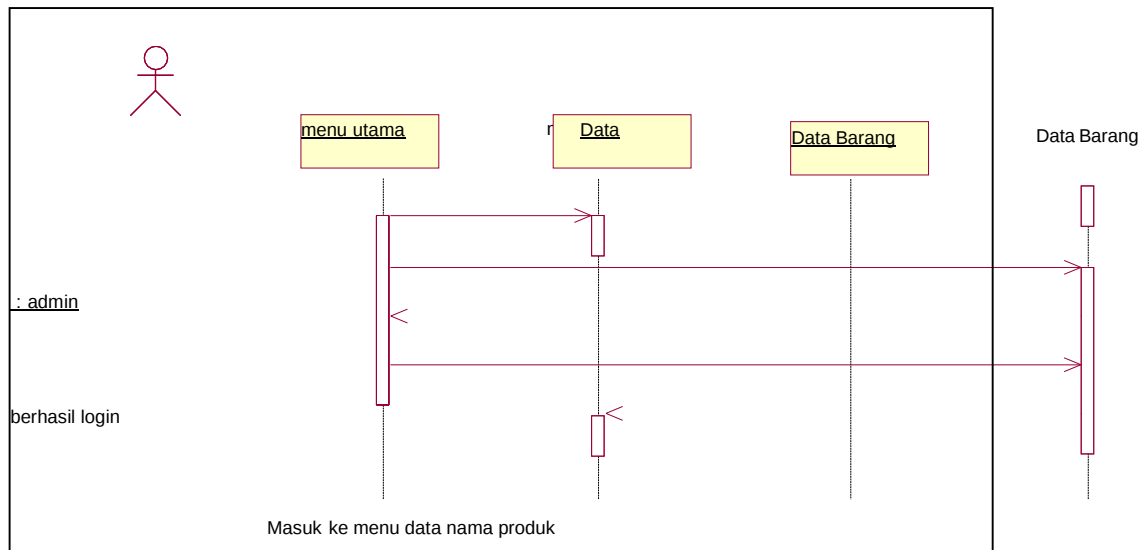
Gambar 4.7 Sequence Diagram Create User



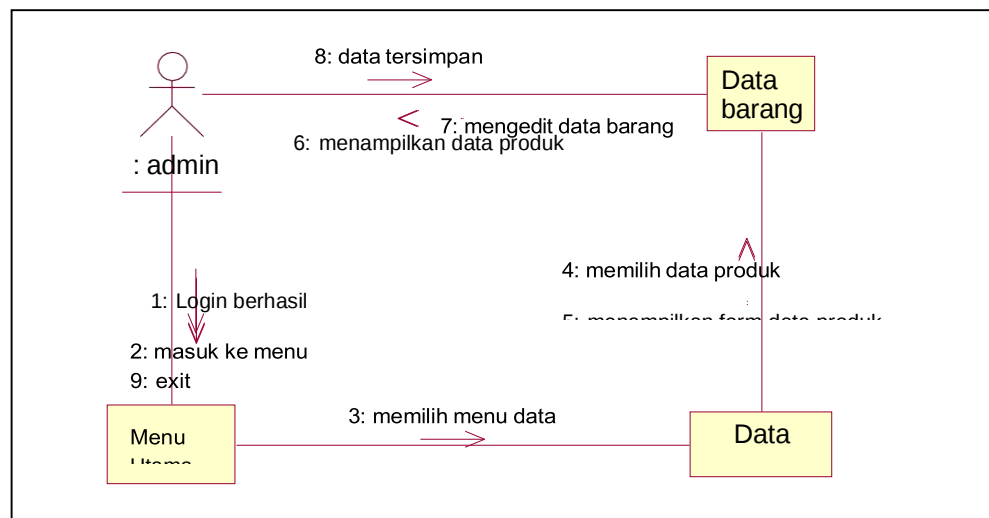
Gambar 4.8 Collaboration Diagram create user

4.6.3 Scenario Sequence Diagram dan Collaboration Diagram Data Barang

Name	Data barang
Aktor	User (Admin)
Goal	ubah produk dan hapus produk berhasil dilakukan.
Precondition	Membuka menu Data
Postcondition	Berhasil masuk ke data barang, mengubah serta menghapus produk.
Step	1. Admin membuka menu data dan milih Data barang 2. Admin dapat melihat barang apa saja yang telah di buat. 3. Admin dapat juga melakukan pengubahan data serta penghapusan data.



Gambar 4.9 Sequence Diagram Nama Produk

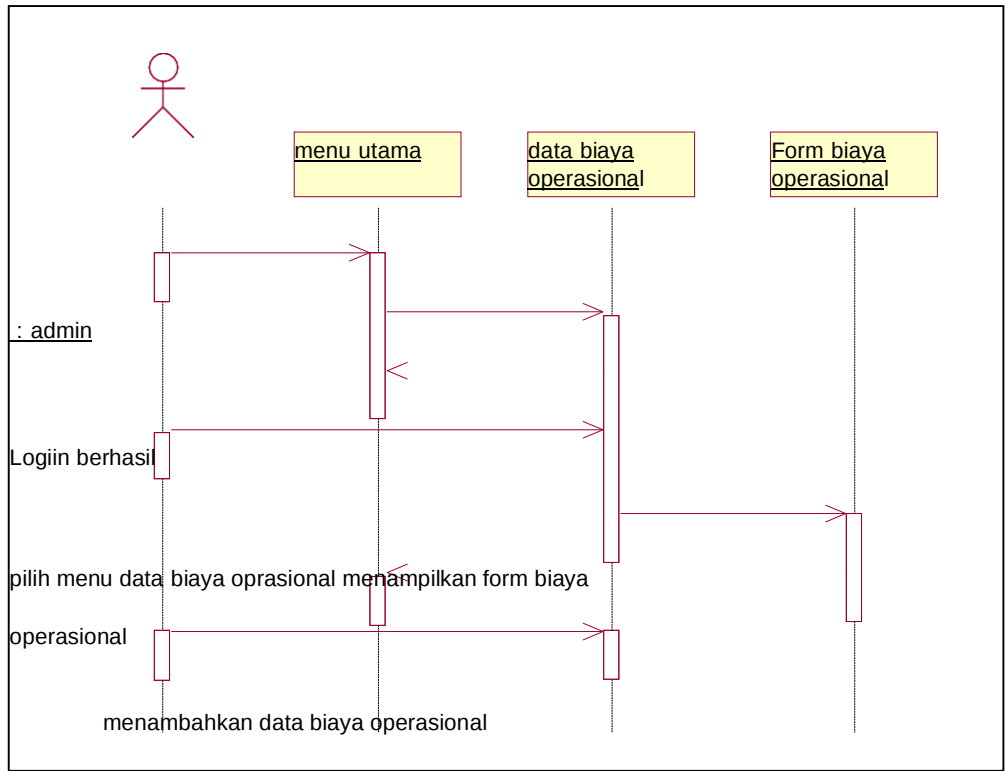


Gambar 4.10 Collaboration Diagram nama Produk

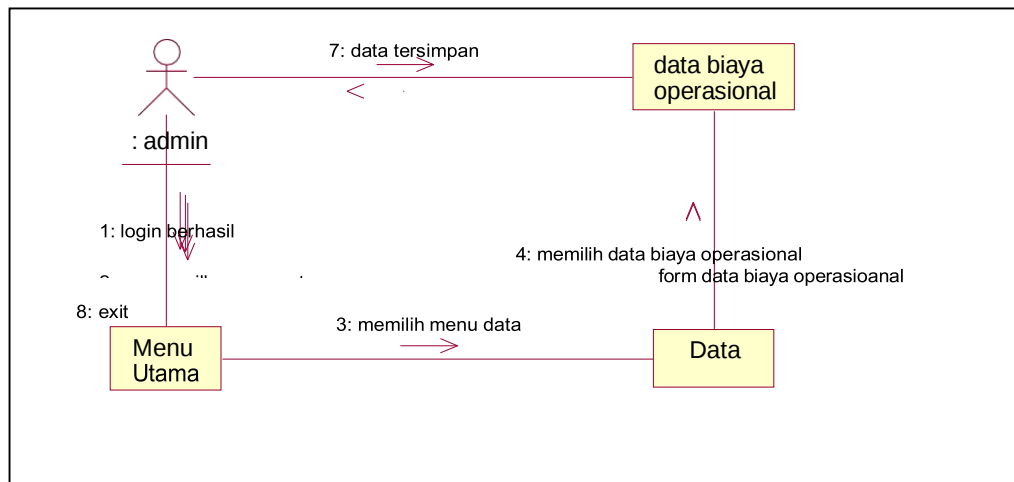
4.6.4 Scenario Sequence Diagram dan Collaboration Diagram Data Biaya Operasional

Name	Data biaya operasional
Aktor	User (admin)
Goal	Input,simpan dan edit data biaya operasional
Precondition	Membuka menu data dan memilih data biaya operasional
Postcondition	Berhasil masuk ke data biaya operasional

Step	1. Admin membuka menu data dan memilih data biaya operasional 2. Admin melakukan penginputan data baru dan dapat mengedit data yang sudah ada
------	--



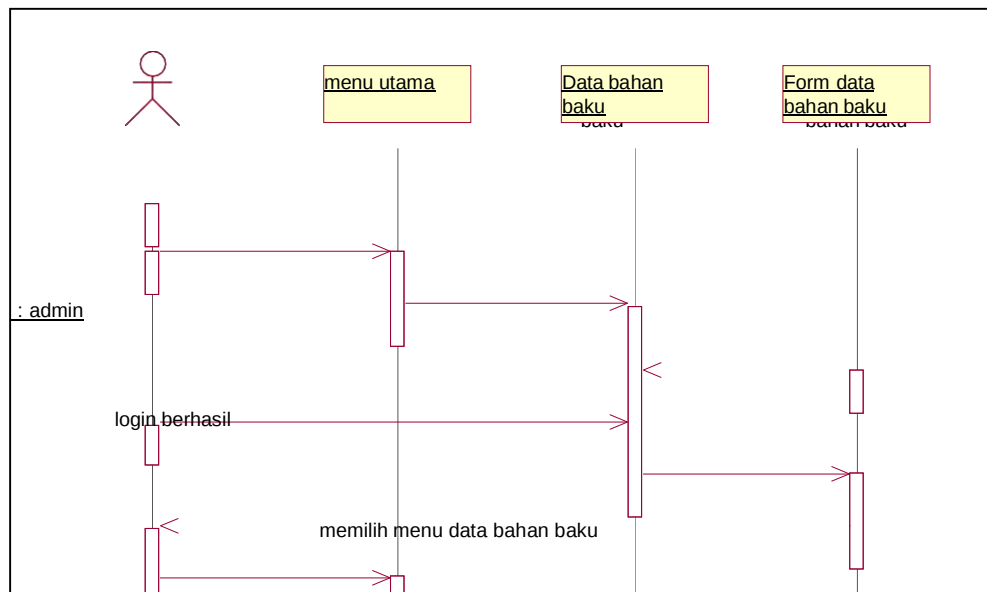
Gambar 4.11 Sequence Diagram Data Biaya Operasional



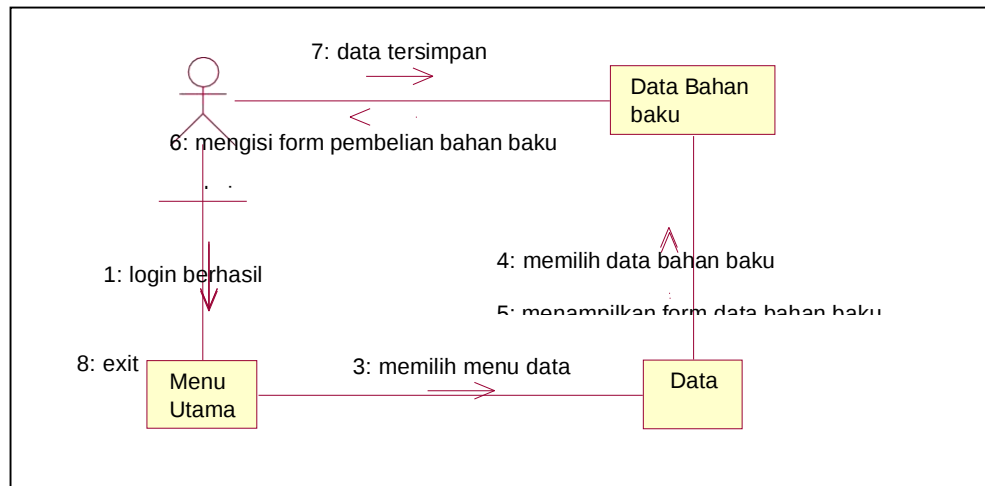
Gambar 4.12 Collaboration Diagram Data Biaya Operasioanal

4.6.5 Scenario Sequence Diagram dan Collaboration Diagram Data bahan baku

Name	Data bahan baku
Aktor	User (admin)
Goal	Input,simpan dan edit nama supplier dan bahan bku
Precondition	Membuka menu data dan memilih data bahan baku
Postcondition	Berhasil menampilkan bahan baku berserta nama supplier
Step	1. Admin membuka menu data dan memilih databahan baku 2. Admin melakukan peng inputan dan menyimpannya.



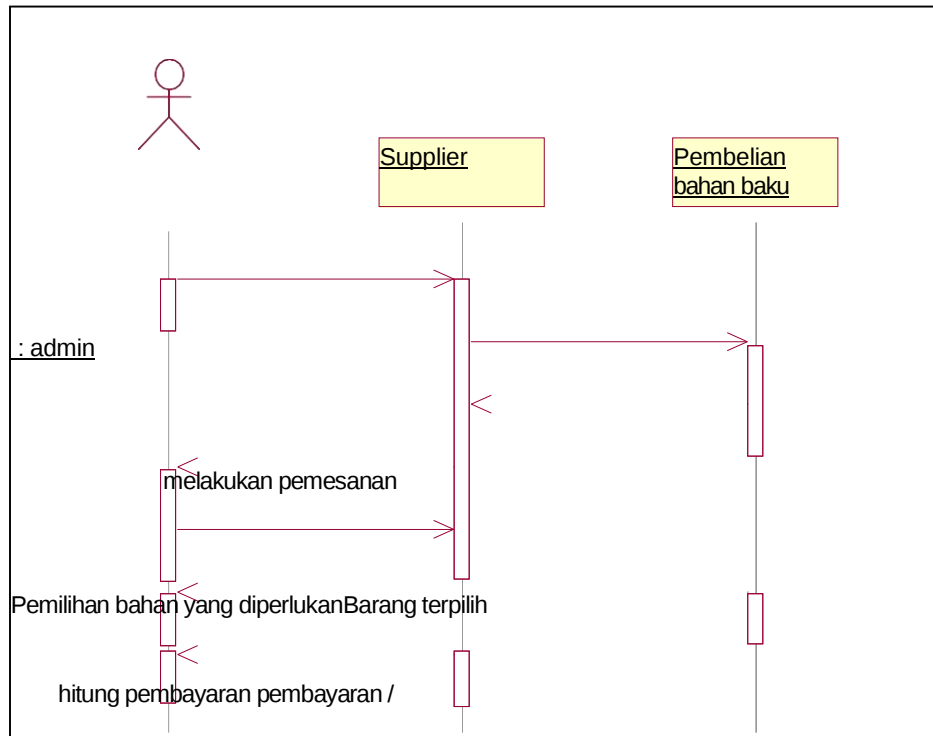
Gambar 4.13 Sequence Diagram Data Bahan Baku



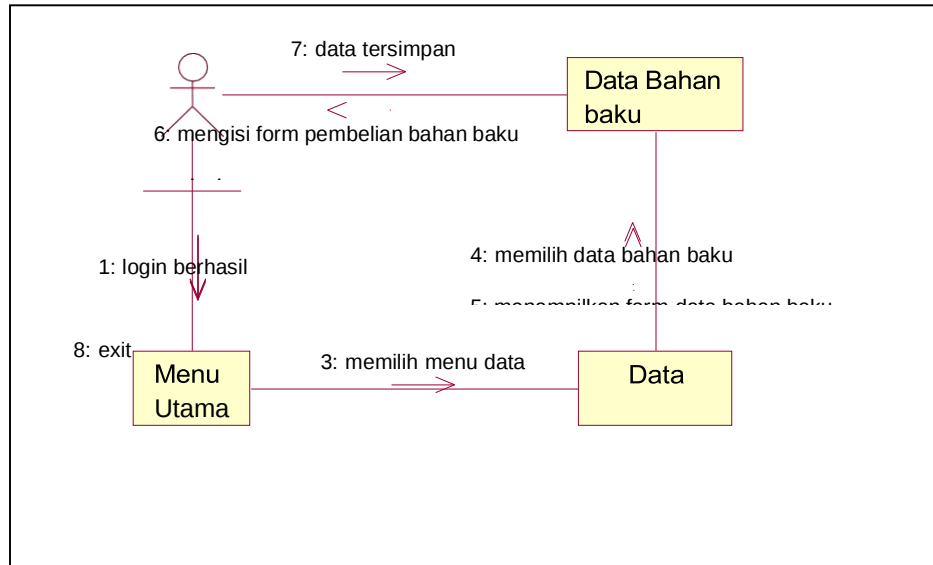
Gambar 4.14 Collaboration Diagram Data Bahan Baku

4.6.6 Scenario Sequence Diagram dan Collaboration Diagram pembelian bahan baku

Name	Pembelian bahan baku
Aktor	User (admin)
Goal	Input,simpan dan edit bahan baku yang akan dibeli
Precondition	Membuka menu entry biaya produksi dan memilih pembelian bahan baku
Postcondition	Berhasil memasukan bahan baku yang akan dibeli
Step	1. Admin membuka menu entry biaya produksi dan memilih pembelian bahan baku 2. Admin melakukan pembelian bahan baku dan menyimpannya.



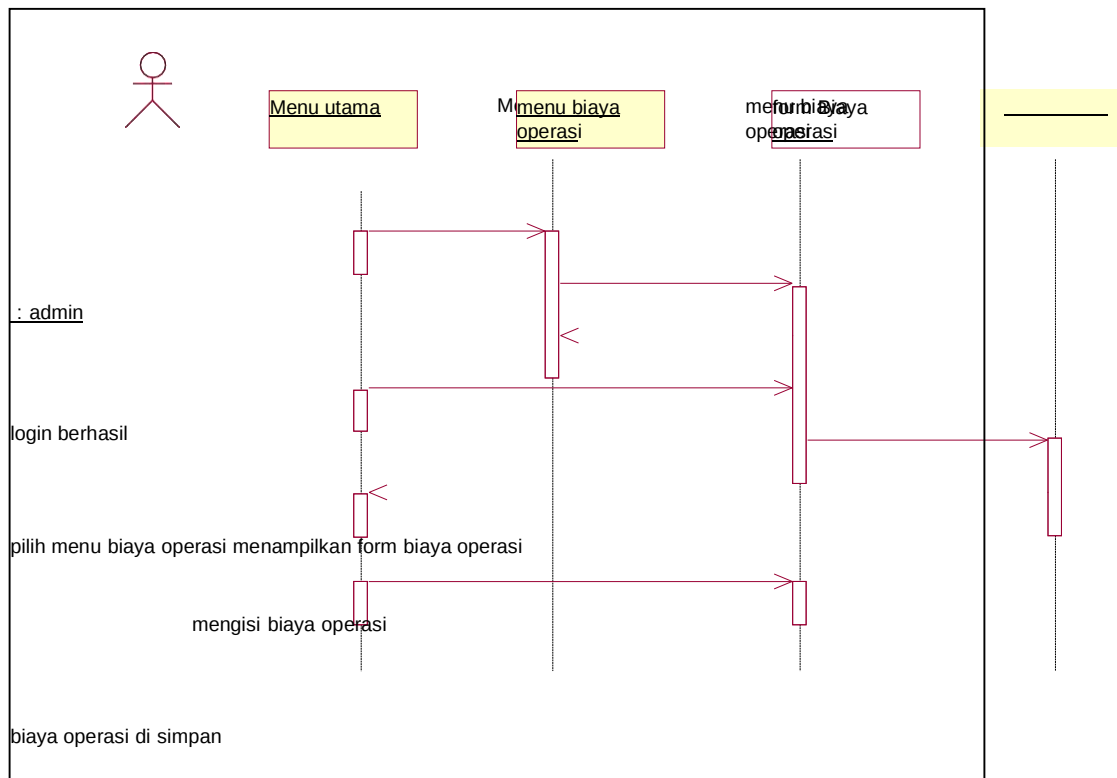
Gambar 4.15 Sequence Diagram pembelian bahan baku



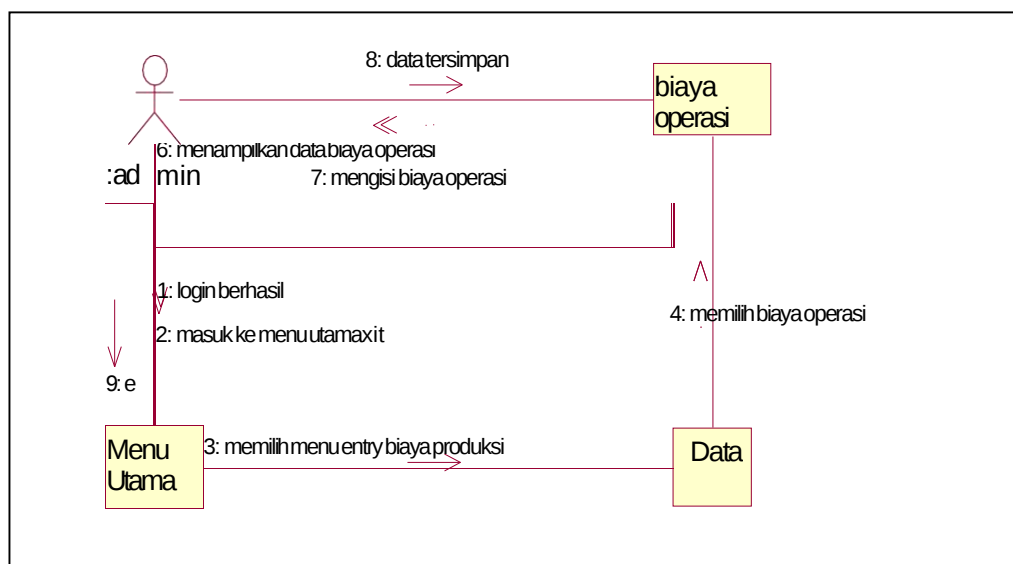
Gambar 4.16 Collaboration Diagram pembelian bahan baku

4.6.7 Scenario Sequence Diagram dan Collaboration Diagram biaya operasi

Name	Biaya operasi
Aktor	User (admin)
Goal	Input,simpan dan edit biaya operasi
Precondition	Membuka menu entry biaya produksi dan memilih biaya operasi
Postcondition	Berhasil mengetahui dan menyimpan biaya operasi apa saja yang dikeluarkan
Step	1. Admin membuka menu data dan memilih biaya operasi 2. Admin memasukkan biaya operasi apa saja yang di keluarkan.3, Admin menyimpan data baru yang telah dimasukkan dengan mengklik simpan, selain itu admin juga dapat melakukan pengubahan data serta penghapusan data.



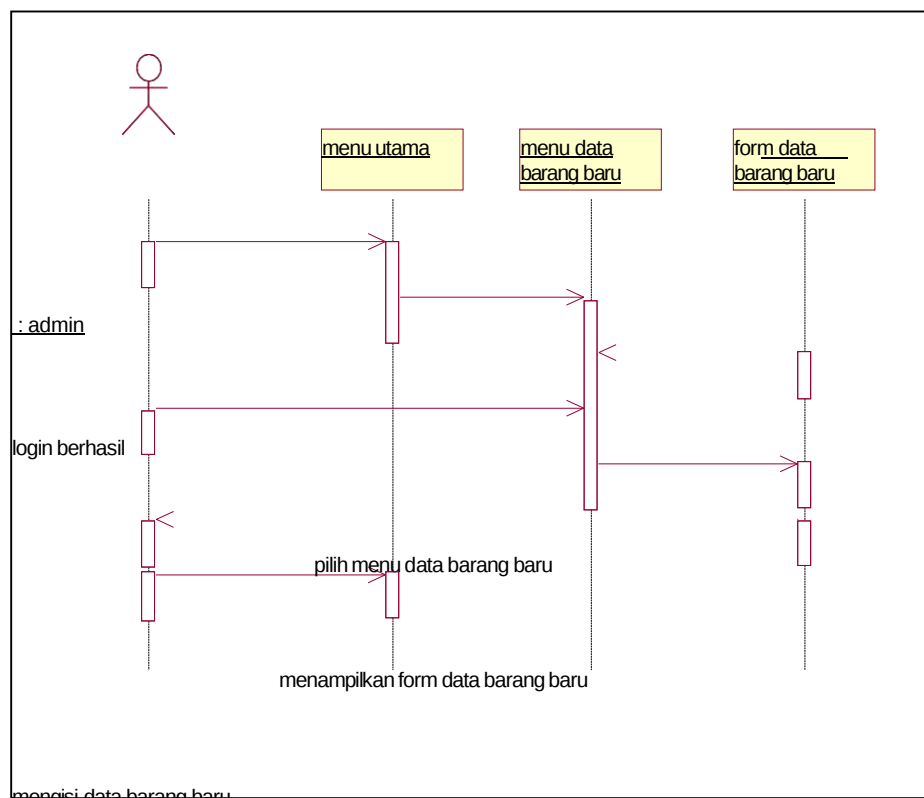
Gambar 4.17 Sequence Diagram biaya operasi



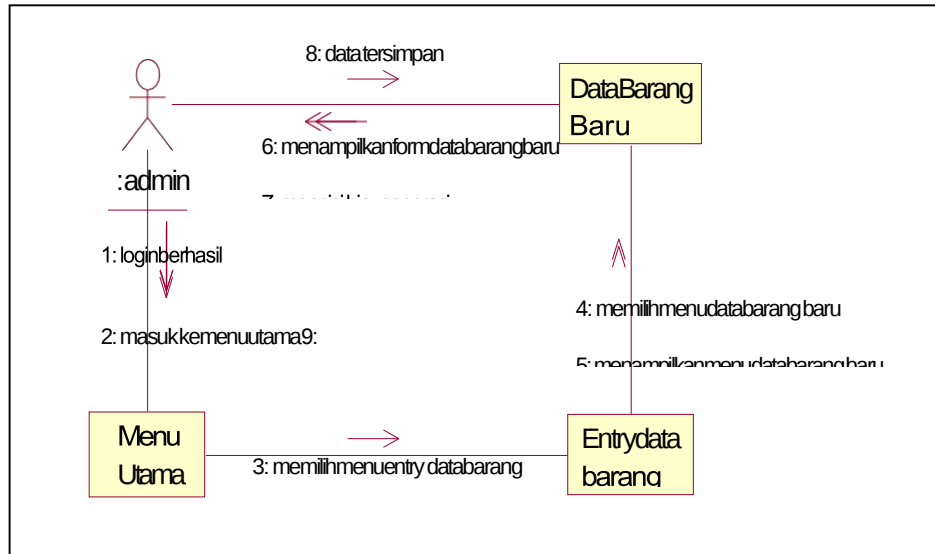
Gambar 4.18 Collaboration Diagram biaya operasi

4.6.8 Scenario Sequence Diagram dan Collaboration Diagram Data barang baru

Name	Data barang baru
Aktor	User (admin)
Goal	Input,simpan dan edit barang baru
Precondition	Membuka menu entry data barang dan memilih data barang baru
Postcondition	Berhasil menyimpan barang baru dan menghitung keuntungan barang yang di inginkan
Step	1. Admin membuka menu entry data barang dan memilih data barang baru 2. Admin memberikan harga mengelola keuntungan dari satu barang tersebut 3. Admin menyimpan data baru yang telah dimasukkan dengan mengklik simpan, selain itu admin juga dapat melakukan perubahan data serta penghapusan data.



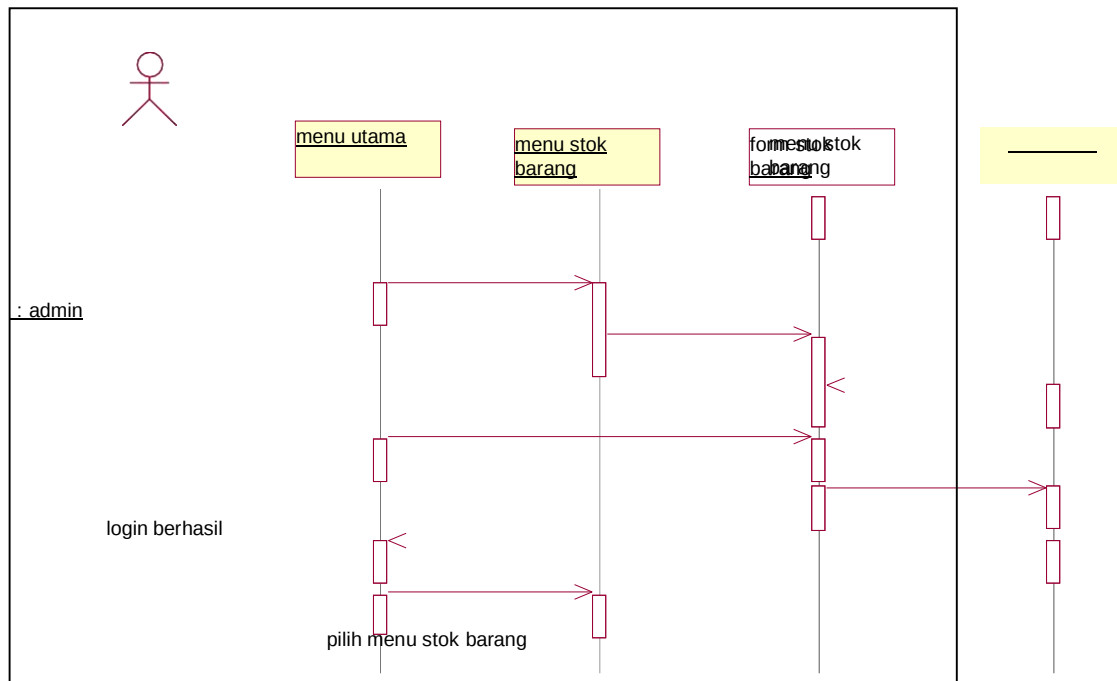
Gambar 4.19 Sequence Diagram data barang barang



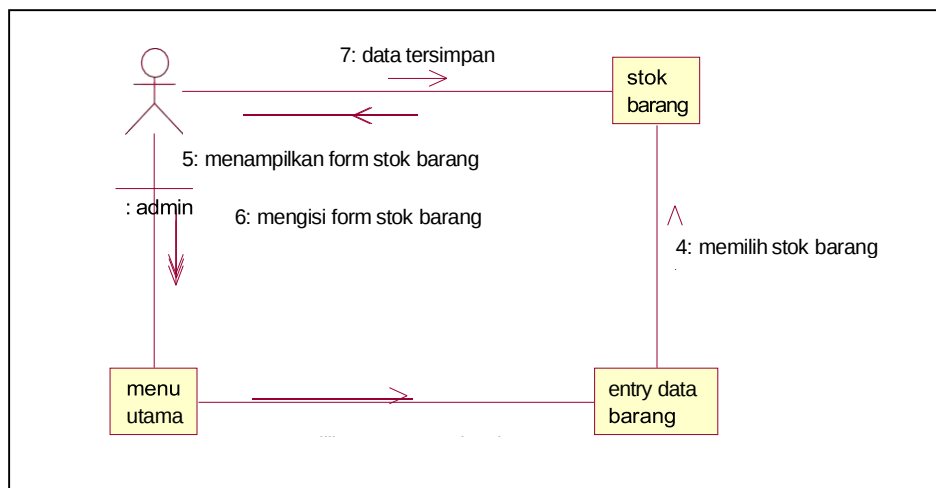
Gambar 4.20 Collaboration Diagram Data barang baru

4.6.9 Scenario Sequence Diagram dan Collaboration Diagram stok barang

Name	Data Stok barang
Aktor	User (admin)
Goal	Input, simpan dan edit stok barang
Precondition	Membuka menu entry data barang dan memilih stok barang
Postcondition	Berhasil melakukan pengecekan stok awal dan stok akhir
Step	1. Admin membuka menu entry data barang dan memilih data stok barang 2. Admin menyimpan data baru yang telah dimasukkan dengan mengklik simpan, selain itu admin juga dapat melakukan pengubahan data serta penghapusan data.



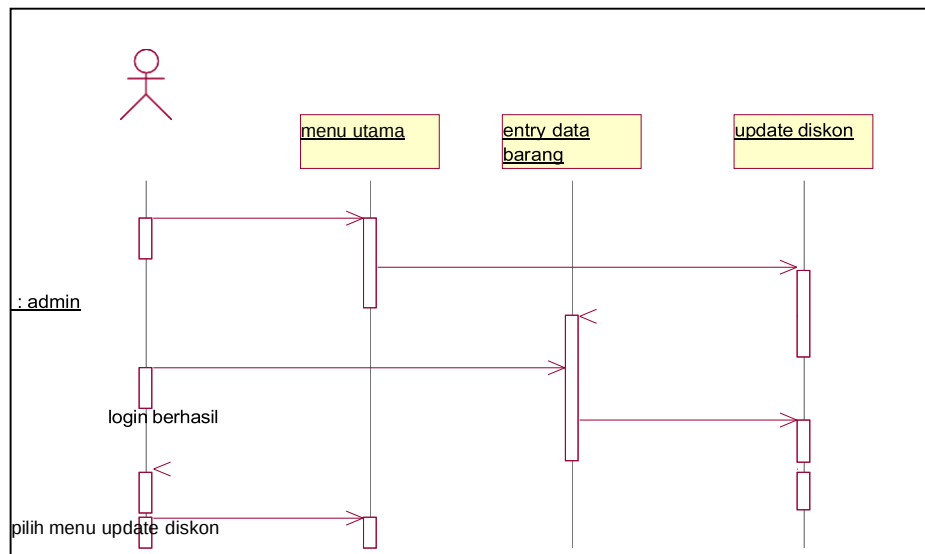
Gambar 4.21 Sequence Diagram stok barang



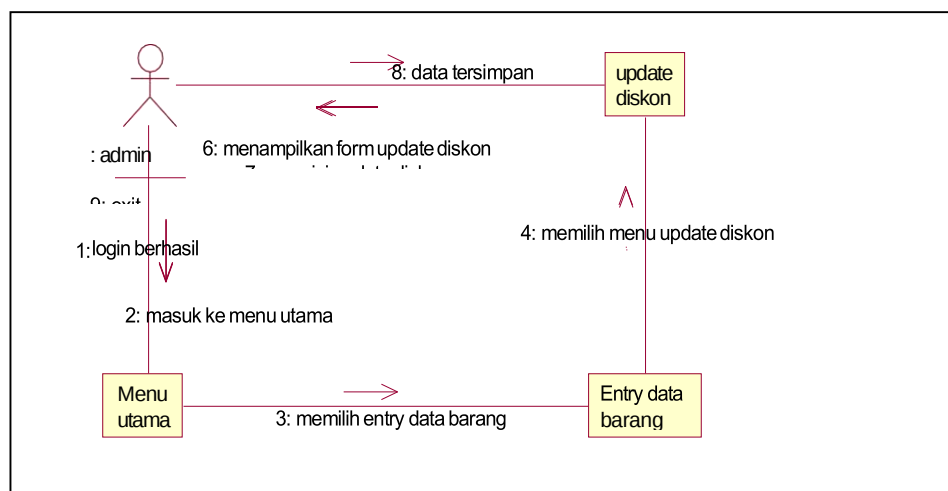
Gambar 4.22 Collaboration Diagram stok barang

4.6.10 Scenario Sequence Diagram dan Collaboration Diagram update diskon

Name	Update diskon
Aktor	User (admin)
Goal	Perubahan diskon
Precondition	Membuka menu entry data barang dan memilih update diskon
Postcondition	Berhasil merubah dan menyimpan diskon dalam pembelian barang yang di tentukan
Step	1. Admin membuka menu entry data barang dan memilih update barang 2. Admin mengupdate diskon sesuai prosedur Dan menyimpannya.



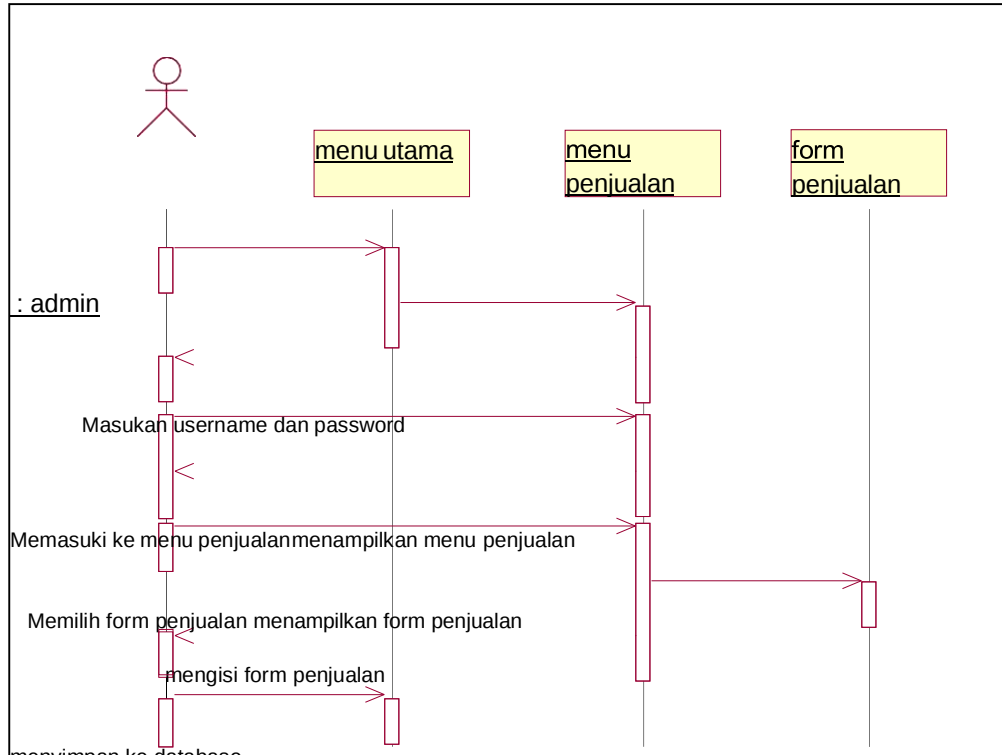
Gambar 4.23 Sequence Diagram update diskon



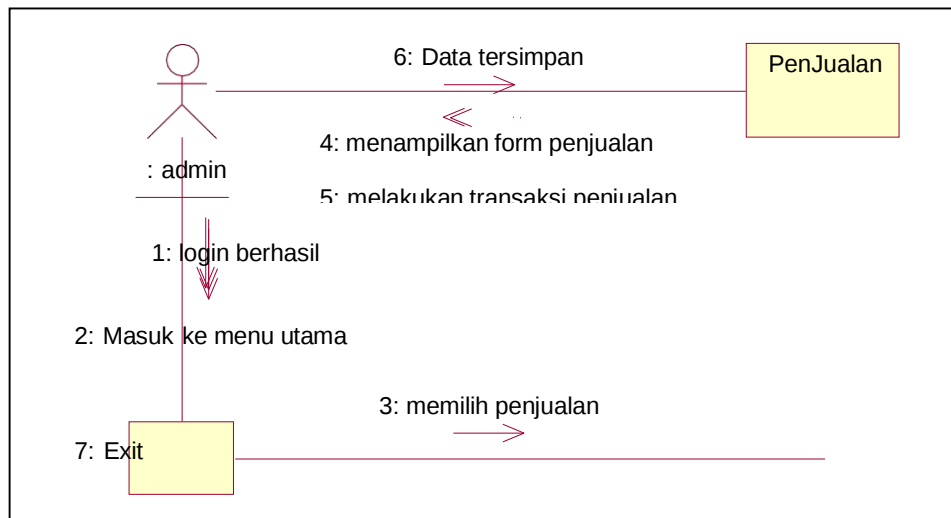
Gambar 4.24 Collaboration Diagram Update Diskon

4.6.11 Scenario Sequence Diagram dan Collaboration Diagram penjualan

Name	penjualan
Aktor	User (admin dan operator)
Goal	melakukan transaksi penjualan
Precondition	Membuka menu penjualan
Postcondition	Berhasil melakukan transaksi penjualan
Step	1. Admin membuka menu penjualan 2. menginput barang apa saja yang akan di beli dan melakukan transaksi penjualan barang



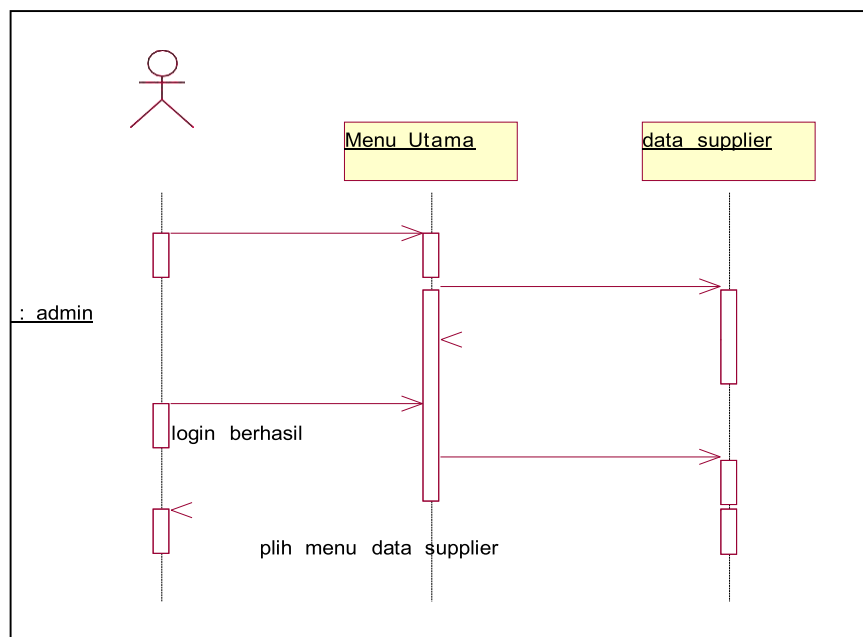
Gambar 4.25 Sequence Diagram penjualan

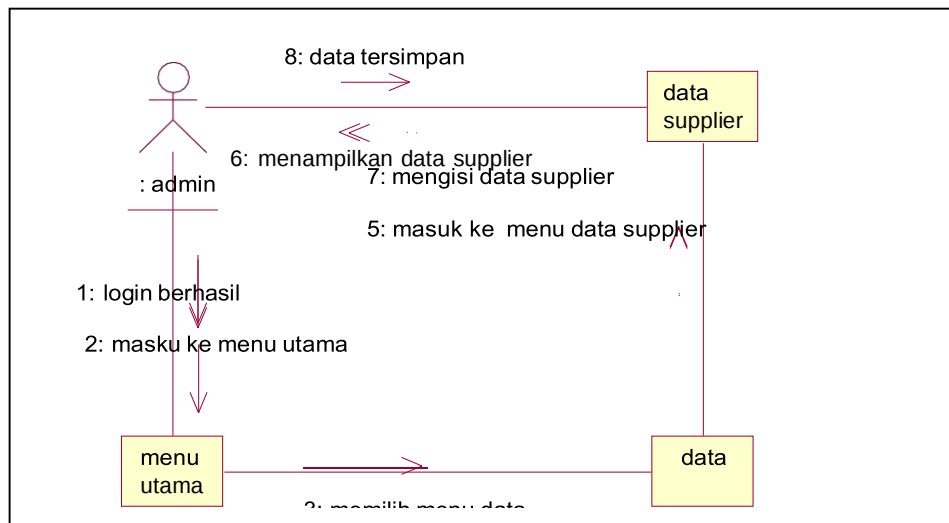


Gambar 4.26 Collaboration Diagram Penjualan

4.6.12 Scenario Sequence Diagram dan Collaboration Diagram Entry data supplier

Name	Entry Data Supplier
Aktor	User (admin)
Goal	Input,edit,hapus data supplier
Precondition	Membuka menu entry data supplier
Postcondition	Berhasil menyimpan data supplier
Step	1. Admin membuka menu entry data supplier 2. menambahkan,edit dan menyimpan data supplier bahan baku

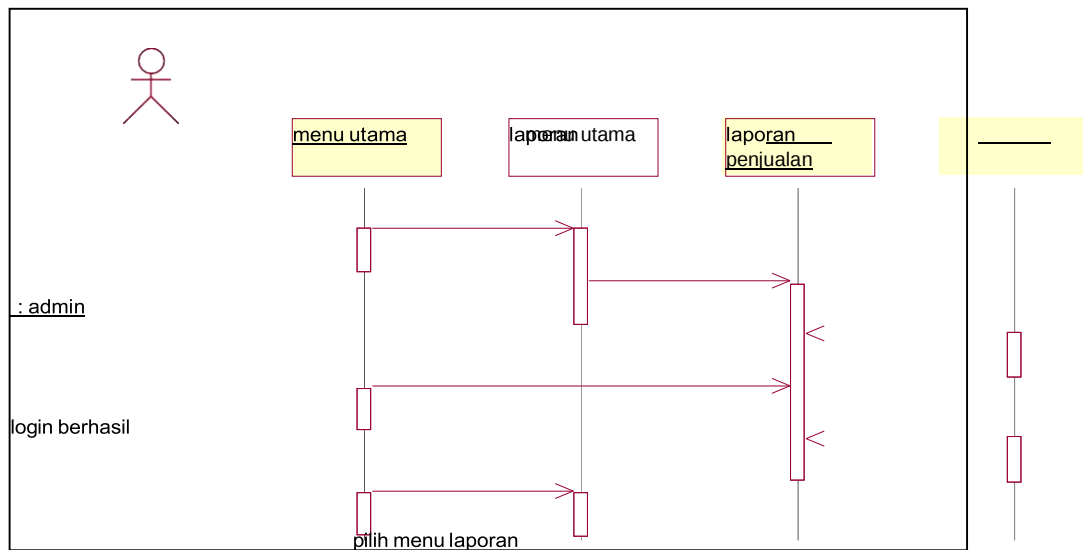




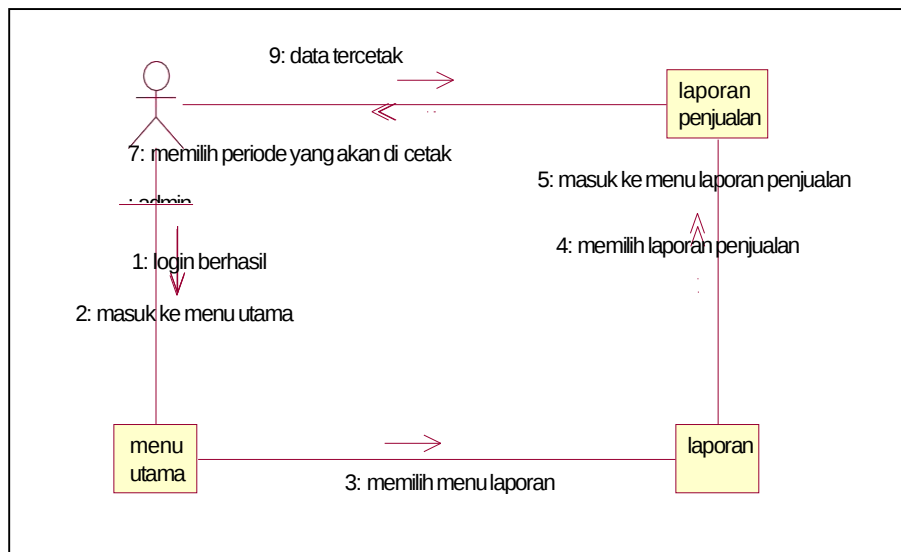
Gambar 4.28 Collaboration Diagram Entry Data Supplier

4.6.13 Scenario Sequence Diagram dan Collaboration Diagram laporan penjualan

Name	Laporan penjualan
Aktor	User (admin dan operator)
Goal	Laporan penjualan dicetak
Precondition	Membuka menu laporan dan memilih laporan penjualan.
Postcondition	Berhasil mencetak laporan penjualan.
Step	1. user membuka menu laporan dan memilih laporan penjualan. 2. user memilih kategori berdasarkan periode 3. user mencetak laporan penjualan berdasarkan periode harian, bulanan dan tahunan



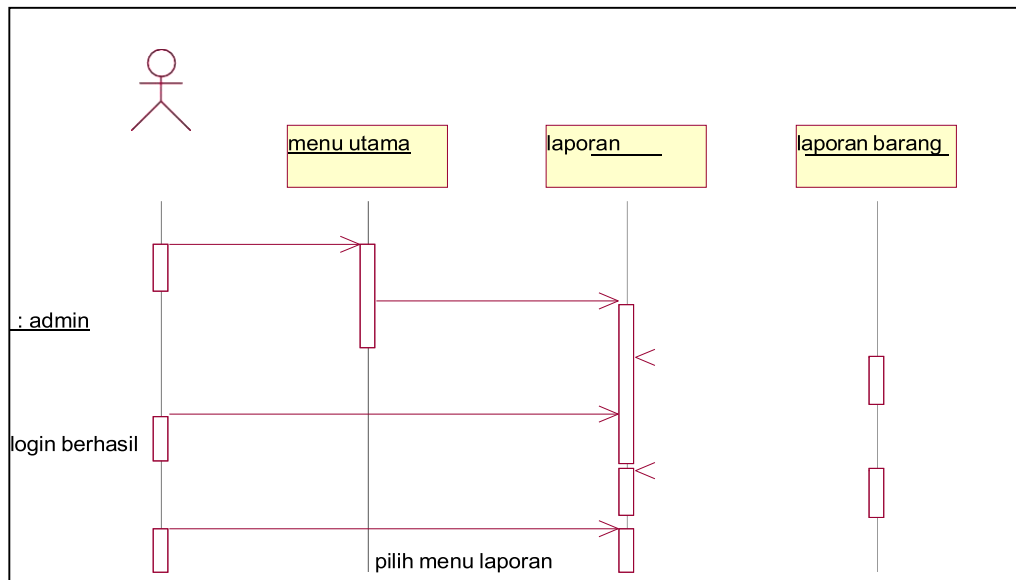
Gambar 4.29 Sequence Diagram update Laporan Penjualan



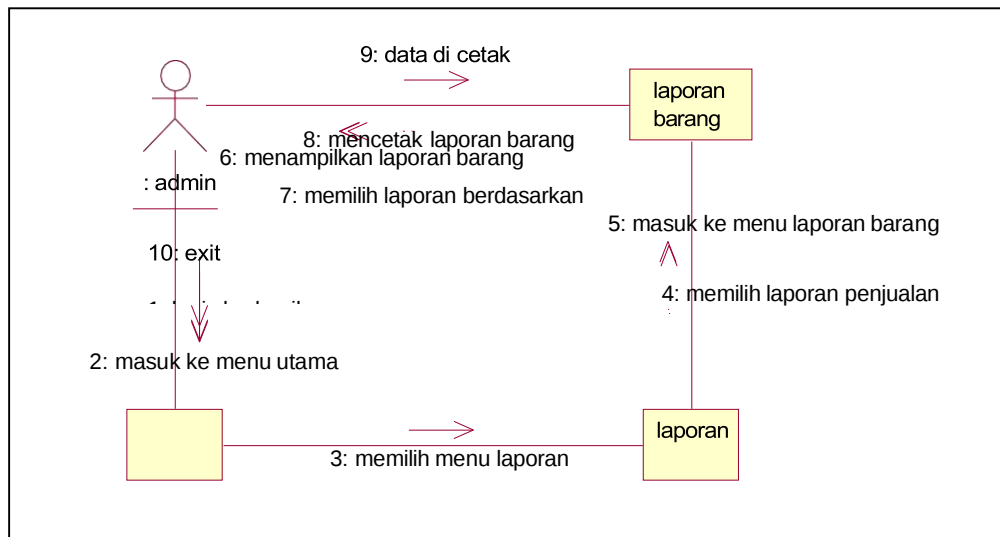
Gambar 4.30 Collaboration Diagram Laporan Penjualan

4.6.14 Scenario Sequence Diagram dan Collaboration Diagram laporan barang

Name	Laporan barang
Aktor	User (admin dan operator)
Goal	Laporan penjualan dicetak
Precondition	Membuka menu laporan dan memilih laporan barang.
Postcondition	Berhasil mencetak laporan barang.
Step	1.user membuka menu laporan dan memilih laporan barang. 2.user memilih dan mencetak laporan barang



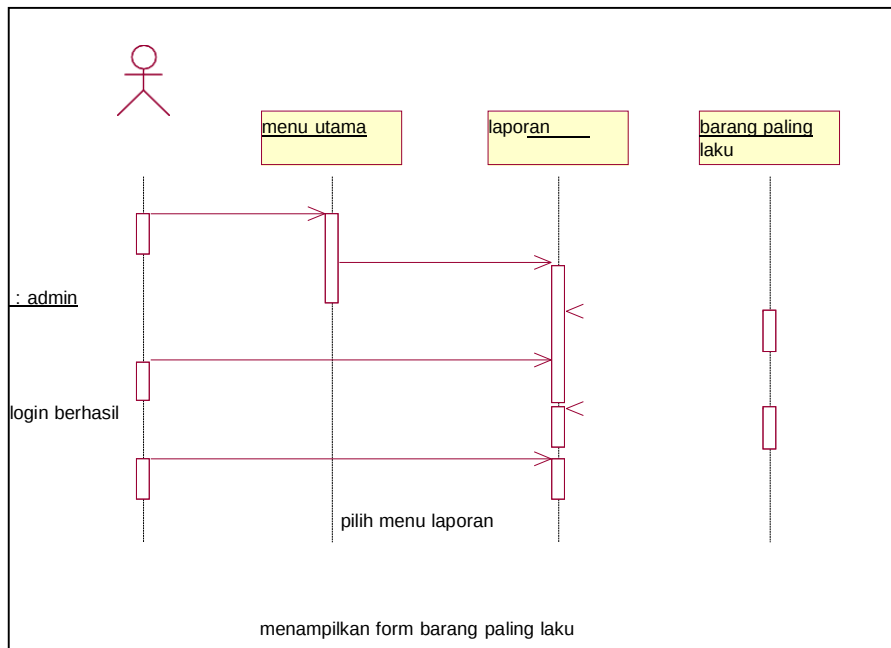
Gambar 4.31 Sequence Diagram Laporan Barang



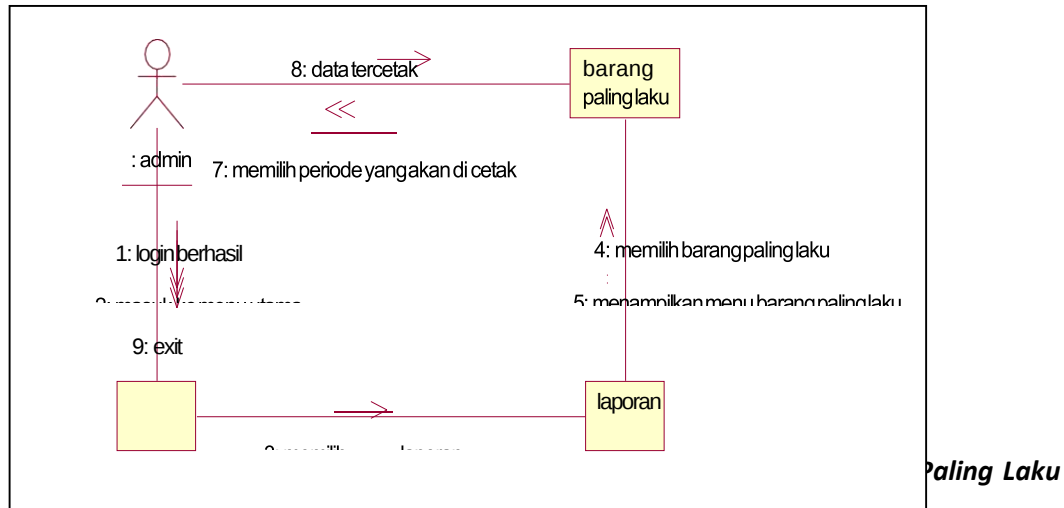
Gambar 4.32 Collaboration Diagram Laporan barang

4.6.15 Scenario Sequence Diagram dan Collaboration Diagram barang paling laku

Name	barang paling laku
Aktor	User (admin dan operator)
Goal	Akan tampil grafik penjualan
Precondition	Membuka menu laporan dan memilih barang paling laku.
Postcondition	Berhasil menampilkan grafik penjualan.
Step	1.user membuka menu laporan dan memilih barang paling laku 2.menampilkan grafik penjualan paling laku



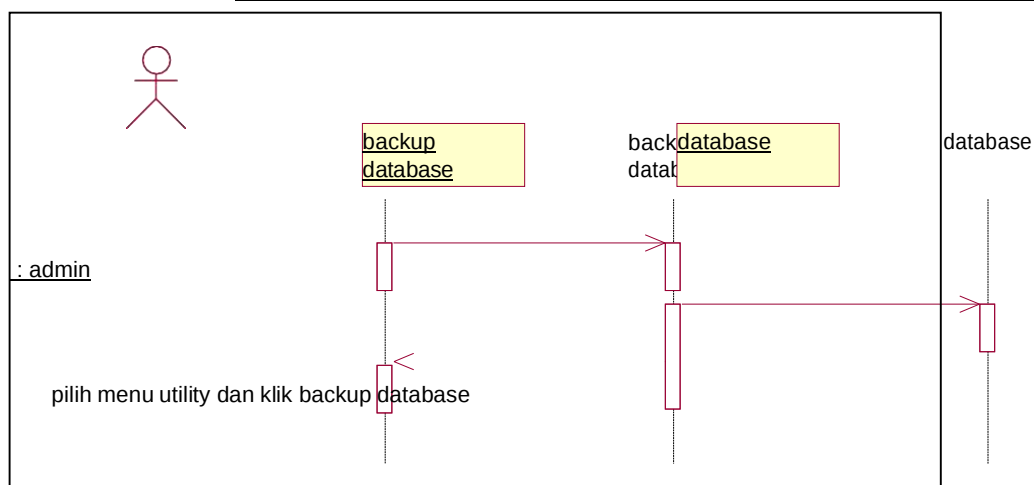
Gambar 4.33 Sequence Diagram Barang paling laku



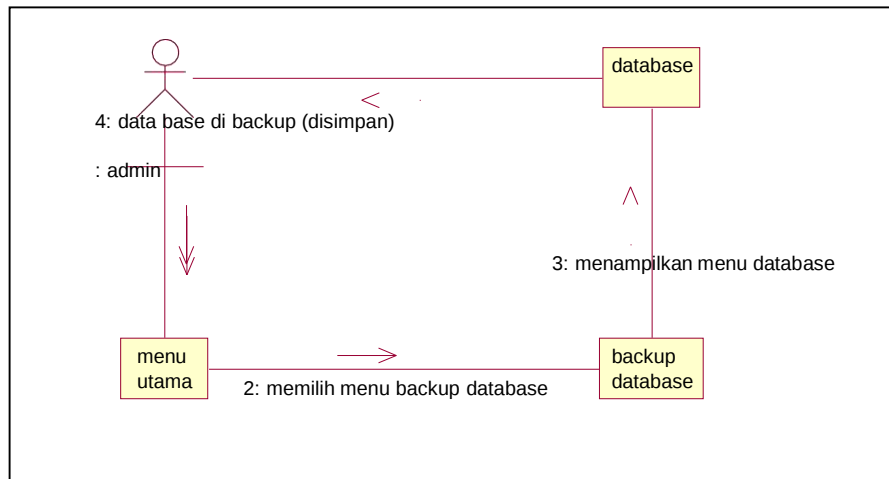
Gambar 4.34 Collaboration Diagram barang paling laku

4.6.16 Scenario Sequence Diagram dan Collaboration Diagram backup database

Name	Backup Database
Actor	User (admin dan operator)
Goal	Data di backup (disimpan)
Precondition	Membuka menu utility dan memilih backup database
Postcondition	Berhasil menyimpan database.
Steps	1. user membuka menu utility dan memilih backup database. 2. user menyimpan database.



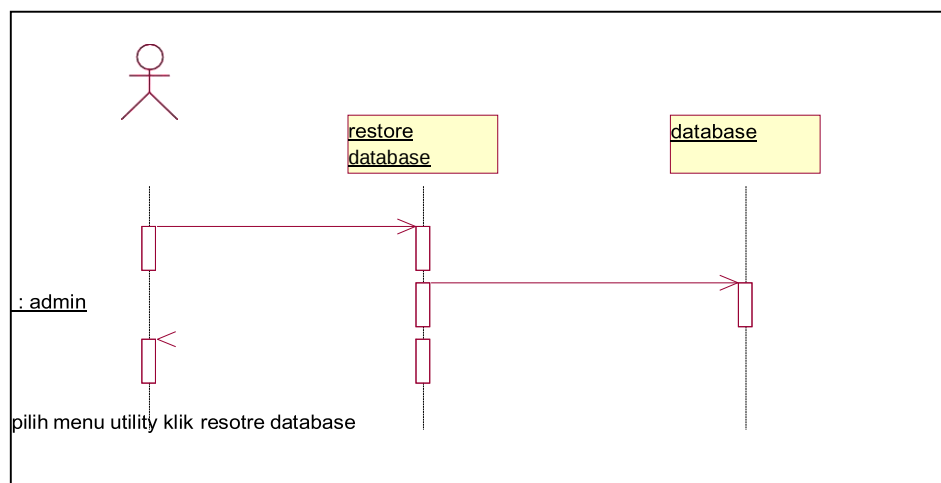
Gambar 4.35 Sequence Diagram Backup database



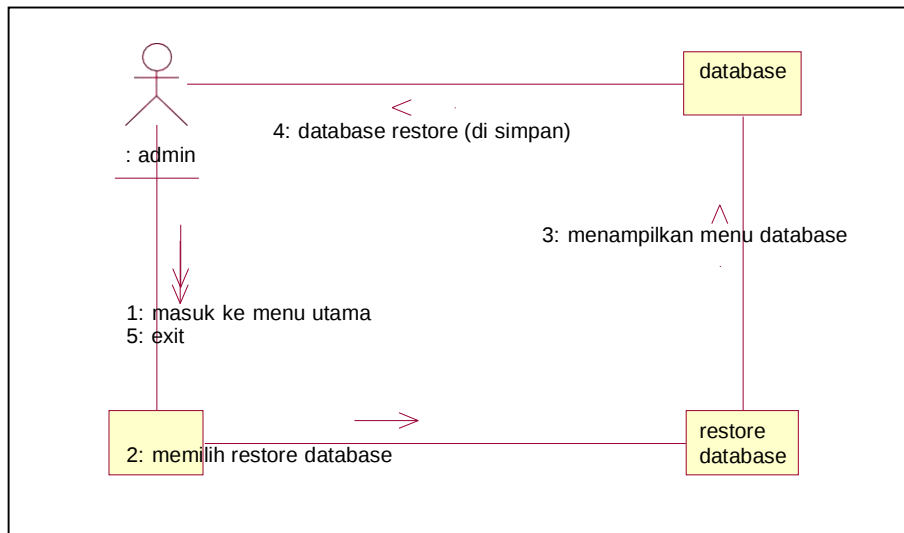
Gambar 4.36 Collaboration Diagram Backup database

4.6.17 Scenario Sequence Diagram dan Collaboration Diagram restore database

Name	Restore Database
Actor	User (admin dan operator)
Goal	Data di restore (diambil)
Precondition	Membuka menu utility dan memilih restore database
Postcondition	Berhasil menyimpan database.
Steps	1. user membuka menu utility dan memilih restore database. 2. User mengambil database.



Gambar 4.37 Sequence Diagram Restore database

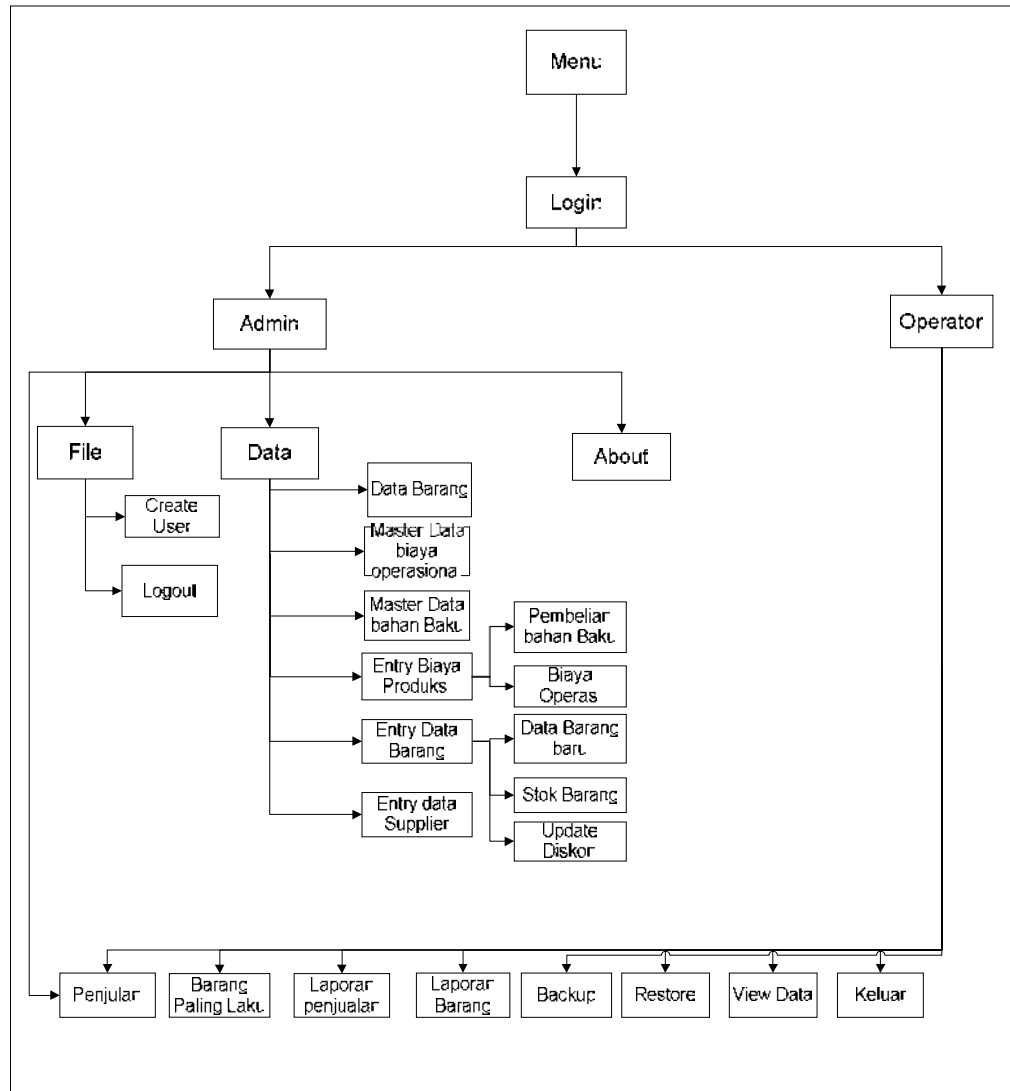


Gambar 4.38 *Collaboration Diagram Restore database*

4.7 Perancangan Menu

Berikut ini merupakan menu Perancangan
digambarkan sebagai berikut :

Aplikasi penjualan di ukm



Gambar 4.39 Perancangan Menu

4.8 Lingkungan Operasional

Lingkungan operasional merupakan kebutuhan-kebutuhan perangkat lunak pada saat diimplementasikan, baik itu perangkat lunak, perangkat keras, maupun karakteristik dari pengguna perangkat lunak tersebut.

4.9 Perangkat Lunak

Perangkat lunak yang dibutuhkan dalam aplikasi penjualan di UKM ini adalah sebagai berikut:

1. Sistem operasi *Windows XP Profesional, Windows Vista, Windows 7*
2. *Visual basis 6.0* sebagai tools pembangun
3. *Microsoft office Access 2007* sebagai tempat penyimpanan *database*.
4. *Rational Rose* untuk membuat atau merancang UML
5. *Cristal report* untuk mencetak laporan
6. *Microsoft Office Visio* sebagai tempat pembuatan layout Aplikasi

4.10 Perangkat Keras

Perangkat keras yang digunakan pada Aplikasi penjualan di UKM ini adalah sebagai berikut:

- a. *Intel Atom*
- b. *Memory 1 GB*
- c. *Hard disk 320 GB*
- d. *Mouse, dan keyboard*

4.11 Karakteristik Pengguna

Kualifikasi dan hak akses yang harus dimiliki pengguna adalah sebagai berikut :

Tabel 4.17 Karakteristik Pengguna Untuk Admin

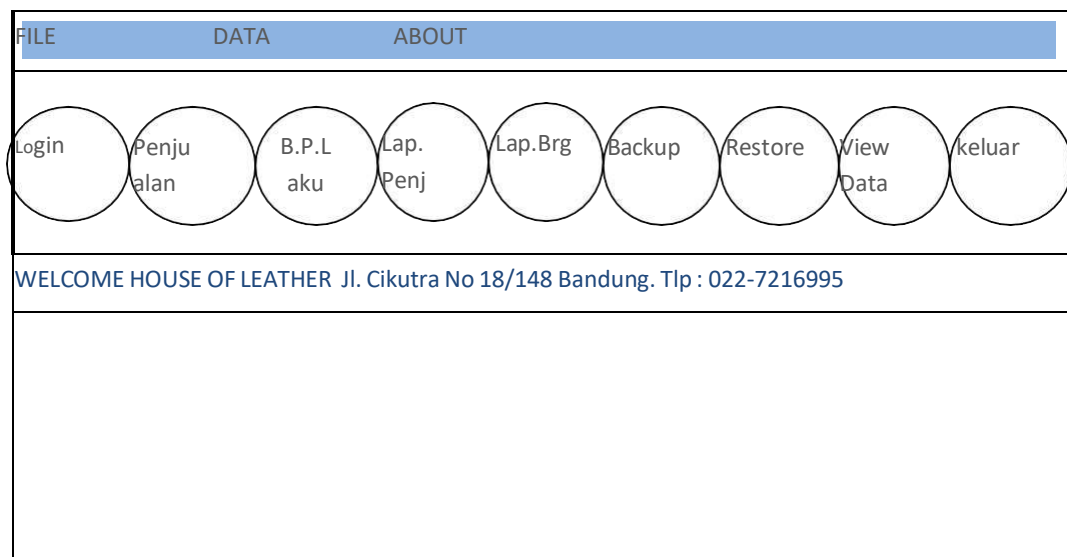
Pengguna	Admin Yang Mengatur Isi Aplikasi
Kualifikasi	1. Mempunyai kemampuan dasar di bidang komputer 2. Dapat mengoperasikan sistem operasi <i>Windows</i>

Tabel 4.18 Karakteristik Pengguna Untuk Operator

Pengguna	Operator Yang Menggunakan Aplikasi
Kualifikasi	1. Mempunyai kemampuan dasar di bidang komputer 2. Dapat mengoperasikan sistem operasi <i>Windows</i>

4.12 Layout Antarmuka

Layout antarmuka merupakan rancangan antarmuka yang akan digunakan sebagai perantara *user* dengan perangkat lunak yang dikembangkan. *Layout* antarmuka dari Aplikasi penjualan adalah sebagai berikut:



Gambar 4.40 Antarmuka Menu Login

MASUKAN ID USER DAN PASSWORD ANDA

ID User

OK

Password

Cancel

House Of Leather

Gambar 4.41 Antarmuka Menu Login

4.12.5 Antarmuka Menu Create User

SILAHKAN CREATE USER DISINI....

Entry Data User

ID User

Username

Password

Ketik Ulang

Pilihan

Data

Simpan

Ubah

Hapus

Batal

Tutup

Tabel User

Gambar 4.42 Antarmuka Create user

4.12.6 Antarmuka Menu Data Barang

TAMBAH DATA PRODUK

Entry Data ProdukID

Produk

Nama Produk

Pilihan

Simpan

Ubah

Hapus

Batal

Tutup

Tabel Data Produk

ID_Produk	Nama
MB001	Dompot
ML001	Tas Hermes Kotak
ML002	Tas Gucci Slim
ML003	Tas Bonia
ML004	Tas Slim Baliy
ML005	Tas Vercase

Gambar 4.43 Antarmuka Data Barang

4.12.7 Antarmuka Menu Master Data Biaya Operasional

ENTRY DATA OPERASIONAL

Entry Data Operasional

ID Operasional

Ket Operasional

Data Baru

Simpan

Ubah

Hapus

Batal

tutup

Pilihan

Tabel Data operasional

ID_BiayaOP

Keterangan

Gambar 4.44 Antarmuka Data Biaya Operasional

4.12.8 Antarmuka Menu Master Data Bahan Baku

ENTRY DATA BAHAN BAKU BARU

Entry Data Bahan baku baru

ID Bahan Baku

ID Supplier Nama

Supplier

Pilihan

Data

Simpan

Ubah

Hapus

Batal

Tutup

Tabel Data Bahan baku

ID BHN.Baku	ID Supplier	Nama	Keterangan
BHN-100	SUP-101	Wawan	Kulit Sapi
BHN-101	SUP-102	Andi	Kulit Buaya
BHN-102	SUP-103	Popo	Kulit Ular
BHN-103	SUP-104	Ida Herdiana	Benang Nilon
BHN-104	SUP-105	Anto	Lem
BHN-105	SUP-106	Kartika	Aksesoris dan Lain-Lain
BHN-106	SUP-107	Sri	

Gambar 4.45 Antarmuka Tambah Data bahan baku

4.12.9 Antarmuka Menu Pembelian Bahan baku

TRANSAKSI PEMBELIAN BAHAN BAKU

Entry Data bahan baku

Kode Bahan

Tanggal Pembelian

Nama Supplier

Keterangan Bahan

Untuk produk

Nama Barang

Jumlah

Total Harga Rp.

View

Pilihan

Data

Simpan

Ubah

Hapus

Batal

Tutup

Tabel Data bahan baku

--	--	--

Gambar 4.46 Antarmuka Pembelian bahan baku

4.12.10 Antarmuka Menu Pembelian Biaya Operasional

BIAYA OPERASIONAL DAN LAIN-LAIN

Entry data Biaya Operasional

Kode Biaya

Biaya produk

Tanggal Keterangan

biayaTotal Biaya

Pilihan

Data

Simpan

Beri

Ubah

Hapus

Batal

Tutup

Tabel data Biaya Operasional

Gambar 4.47 Antarmuka Biaya operasional

4.12.11 Antarmuka Menu Data Barang baru

ENTRY DATA BARANG BARU

Entry Data Barang

Nama Produk

Tanggal

Kode barang

Nama

Stok Awal

Keuntungan / barang Rp.

Harga /Barang Rp.

Biaya Bahan Baku Rp.

Biaya Operasional Rp.

Total Biaya produksi

Harga pokok / Barang

Pilihan

Simpan

Ubah

Hapus

Batal

Tutup

Tabel data Barang

Gambar 4.48 Antarmuka Data barang baru

4.12.12 Antarmuka Menu Stok Barang

TAMBAH STOK BARANG

Entry Data Stok barangID

Barang

View

Nama Barang

Tambah Stok

Pilihan

Simpan

Ubah

Hapus

Tutup

Tabel stok barang

--	--	--

Gambar 4.49 Antarmuka Stok Barang

4.12.13 Antarmuka Menu Update Diskon

ENTRY DATA DISKON

Entry data barang

Mendapatkan Diskon

(%)

Tanggal Update

Pilihan

Update Diskon

Batal

Tutup

Tabel Data Diskon

--	--	--

Gambar 4.50 Antarmuka Update Diskon

4.12.14 Antarmuka Menu Penjualan

TRANSAKSI PEMBELIAN BARANG						
Entry data Barang						
No Faktur					TOTAL HARGA Rp.	
Id Barang Nama						
barang						
Harga / barang Rp.						
Jumlah Barang Harga					TOTAL BAYAR Rp.	
Barang Rp.						
					ADD	
Barang Pilihan						CASH
No barang	Id	Nama Barang	Harga barang	Jum	Harga Barang	BAYAR
						KEMBALIAN

Gambar 4.51 Antarmuka Penjualan

4.12.15 Antarmuka Menu Entry Data Supplier

ENTRY DATA SUPPLIER

Entry Data Supplier

No supplier

Nama Supplier

Kota

Telepon

Keterangan

Pilihan

Gambar 4.52 Antarmuka Entry Data Supplier

4.12.16 Antarmuka Menu Laporan Penjualan

LAPORAN PENJUALAN

Pilih Berdasarkan

☐ PERHARI

☐ PERBULAN

☐ PERTAHUN

Dari Tanggal

Sampai Tanggal

Print View

Tutup

Gambar 4.53 Antarmuka Laporan Penjualan

4.12.17 Antarmuka Menu Laporan Barang

The screenshot shows a window titled "LAPORAN BARANG". Inside, there is a section labeled "Pilih Berdasarkan" with two radio buttons. The first radio button is selected and is labeled "ID Barang Print". The second radio button is labeled "semua". Below this section is a text input field labeled "Kata Kunci". At the bottom of the window, there is a table with three columns and one row.

--	--	--

Gambar 4.54 Antarmuka Laporan Barang

4.12.18 Antarmuka Menu Barang paling laku

The screenshot shows a window titled "LAPORAN PENJUALAN". Inside, there is a section labeled "Pilih Berdasarkan" with two radio buttons. The first radio button is selected and is labeled "PERBULAN". The second radio button is labeled "PERTAHUN". Below this section, there are two date input fields. The first is labeled "Dari Tanggal" and the second is labeled "Sampai Tanggal". At the bottom of the window, there are two buttons: "Print View" and "Tutup".

Gambar 4.55 Antarmuka Barang paling laku

4.12.19 Antarmuka Menu Backup Database

BACKUP DATABASE

Cari lokasi untuk backup database

Cari

Backup database

Nama lokasi backup

Backup

Tutup

Gambar 4.56 Antarmuka Backup Database

4.12.20 Antarmuka Menu Restore Database

RESTORE DATABASE

Cari lokasi database yang sudah di backup untuk melakukan restore

Backup database

Lokasi database di Backup

Cari

Restore

Tutup

Gambar 4.57 Antarmuka Restore Database

4.12.21 Antarmuka Menu View Data

DATA PENCARIAN

Pilih Berdasarkan

☒ Data Barang

☐ Data Supplier

☒ Stok barang

☐ Data Penjualan

BATAL

Kata Kunci

Gambar 4.58 Antarmuka View Data

4.12.22 Antarmuka Menu About

INFO

TENTANG HOUSE OF LEATHER

Ada satu tempat yang cukup
menarik buat beli tas kulit

HOUSE OF LEATHER! tas disini keren – keren ga kalah
deh sama produk - produk luar negeri!

ALAMAT : Jl. Cikutra No 18/148 Bandung. Tlp : 022-7216995

Programming : Karina Dewi A

Gambar 4.59 AntarmukaAbout

IMPLEMENTASI

Bab ini berisi uraian mengenai tahapan untuk membangun/mewujudkan rancangan sistem baru secara nyata. Implementasi sistem merupakan tahapan pengkodean (*coding*) berdasarkan rancangan sistem yang telah dibuat kedalam bahasa pemrograman. Lingkungan implementasi meliputi lingkungan perangkat keras (*hardware*) dan lingkungan perangkat lunak (*software*).

5.1 Implementasi

Penjelasan yang meliputi langkah-langkah pembuatan Aplikasi Penjualan di UKM ke penggunaan aplikasi tersebut. Berikut ini aktifitas yang dilakukan dalam merancang dan mengimplementasikan aplikasi penjualan :

a. Persiapan Instalasi Perkakas (*Tools*) Yang Akan Digunakan

Tahap awal yang dilakukan dalam rangka instalasi perkakas yang dibutuhkan untuk membuat aplikasi penjualan di UKM adalah sebagai berikut.

Tabel 5.1

Spesifikasi Perangkat untuk Pembuatan Aplikasi Penjualan

No	Uraian Perangkat Aplikasi Penjualan
1	Menginstal Visual basic versi 6.0

b. Pengetesan Program Pada Perangkat Lunak

Tahap ini dilakukan untuk menghindari kesalahan antara lain :

- Kesalahan penulisan *source code* program
- Kesalahan saat program yang dijalankan
- Kesalahan logika

c. Evaluasi Perangkat Lunak

Tahap ini diperlukan untuk memastikan apakah perangkat lunak dapat berjalan dengan baik, serta mengidentifikasi *error* sehingga perbaikan dapat segera dilakukan sampai perangkat lunak benar-benar siap guna.

d. Implementasi Aplikasi

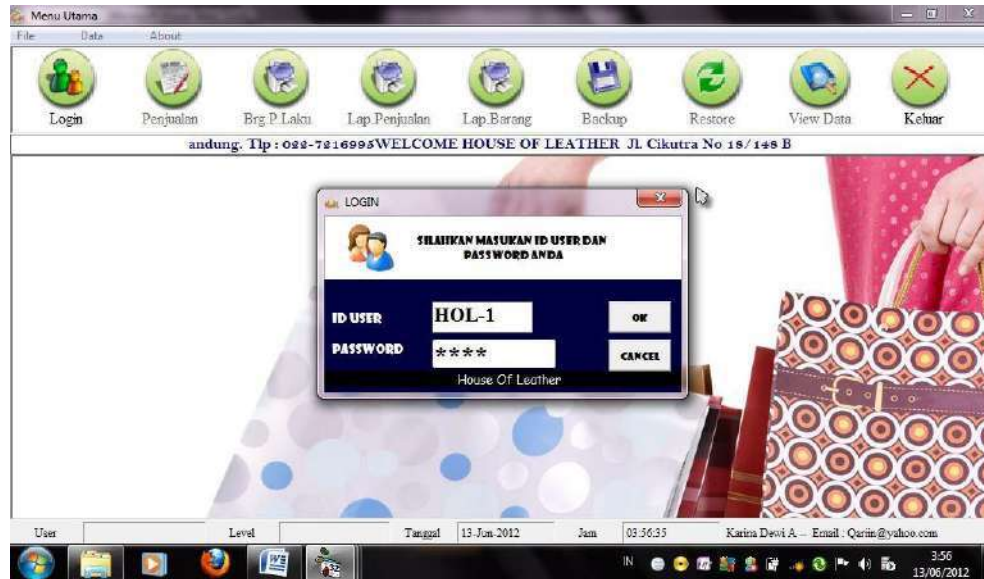
Adapun yang digunakan untuk mengimplementasikan aplikasi Penjualan antara lain :

1. Perangkat Komputer (PC)
2. Monitor
3. Keyboard
4. Mouse

5.2 Implementasi Antarmuka

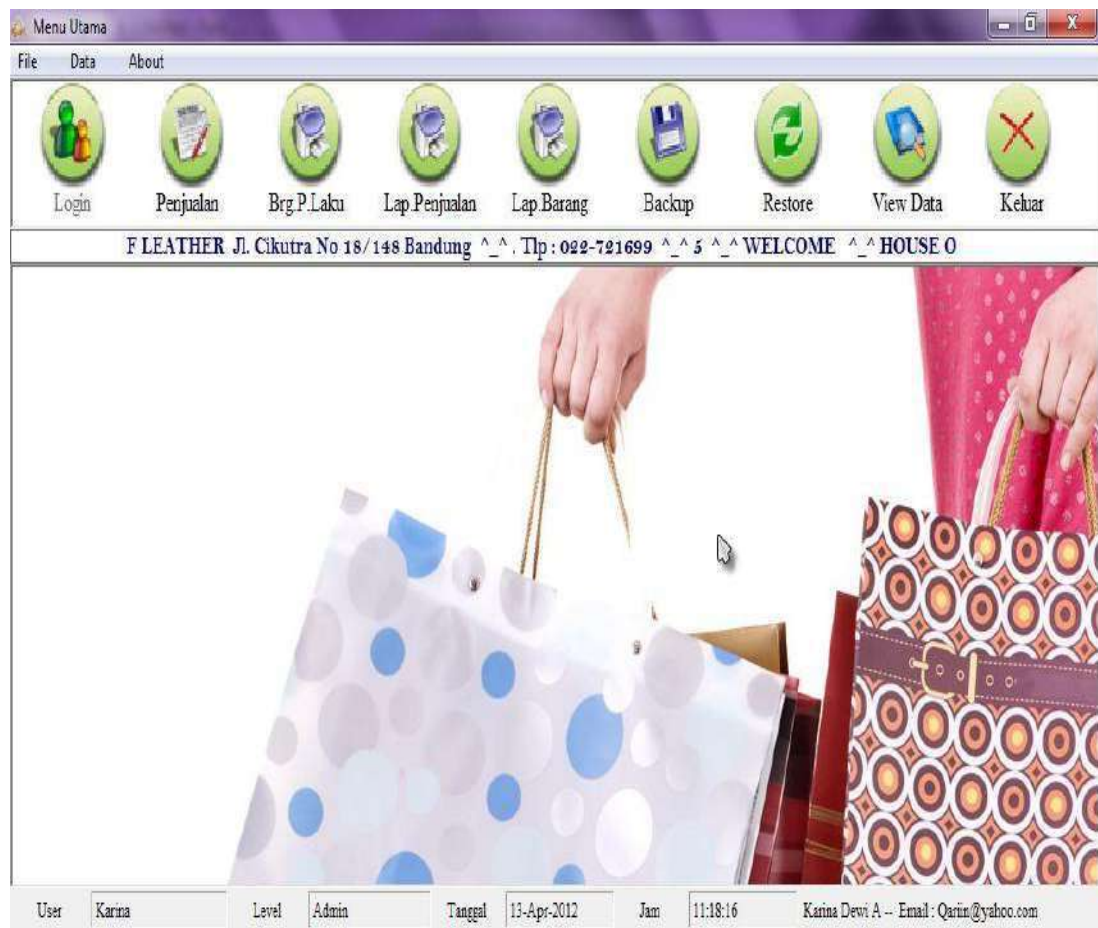
Implementasi rancangan antarmuka dengan menggunakan bahasa pemrograman *Visual Basic 6.0*, tahapan yang harus dilakukan untuk instalasi perangkat lunak yang dihasilkan, mulai dari tahapan persiapan instalasi sampai dengan perangkat lunak siap digunakan beserta petunjuk umum penggunaan perangkat lunak yang digambarkan pada *dialog screen*.

5.2.1 Antarmuka *Login* untuk menampilkan form Login



Gambar 5.1 Antarmuka Login

5.2.2 Antarmuka Form Menu Utama



Gambar 5.2 Antarmuka Menu Utama

5.2.3 Antarmuka Form Create User

ENTRY DATA USER

ID User

Username

Password

Ketik Ulang

Posisi

Aktif:

TABEL USER

ID User	Username	Password	Level
HOL-1	Karna	1111	Admin
HOL-2	Awan	2222	Operator
HOL-3	Zo	3333	Operator
HOL-4	Hedi	4444	Operator

Click tabel user dua kali untuk ubah / hapus data user

Gambar 5.3 Antarmuka Create User

5.2.4 Antarmuka Form Data Barang

PRODUK UTAMA PERUSAHAAN

DATA PRODUK

ENTRY DATA BARANG

ID Barang

Nama Barang

Pilihan:

 **Simpan**  **Ubah**  **Hapus**  **Batal**  **Tutup**

TABEL DATA PRODUK

	ID Produk	Nama
▶	MD001	Dompot
	MT001	Tas Hermes Kota
	MT002	Tas Gucci Slim
	MT003	Tas Slim Baliy
	MT201	Tqs

Click tabel data produk dua kali untuk ubah / hapus data produk

Gambar 5.4 Antarmuka Data Barang

5.2.5 Antarmuka Form Data biaya Operasional

DATA BIAYA OPERASIONAL

ENTRY DATA OPERASIONAL

ID Operasional

Ket Operasional

Pilihan :

Data Baru **Simpan** **Ubah** **Hapus** **Batal** **Tutup**

TABEL DATA OPERASIONAL

ID BiayaOp	Keterangan
OPS-1	Karyawan
OPS-1	Listrik
OPS-1	Telpon
OPS-1	Air
OPS-1	dan lain-lain

Click tabel operasional baku dua kali untuk ubah / hapus data

Gambar 5.5 Antarmuka Data Biaya Operasional

5.2.6 Antarmuka Form Data Bahan Baku

DATA BAHAN BAKU BARU

ENTRY DATA BAHAN BAKU

ID Bahan Baku

ID Supplier

Nama Supplier

Keterangan Barang

Pilihan :

Data Baru **Simpan** **Ubah** **Hapus** **Batal** **Tutup**

TABEL DATA BAHAN BAKU

ID BahanBaku	ID Supl	Nama	Keterangan	
BHN-100	Sup-101	wawan	Kulit Sapi	
BHN-101	Sup-102	andi	kulit buay	
BHN-102	Sup-103	Popo	Kulit Ular	
BHN-103	Sup-104	ida herdia	Benang nil	
BHN-104	Sup-105	hartono	Lem	
BHN-105	Sup-106	kartika	Aksesoris	
BHN-106	Sup-107	Sri	Kain Lapis	
BHN-107	Sup-108	anto	resleting	

Click tabel bahan baku dua kali untuk ubah / hapus data

Gambar 5.6 Antarmuka Data Bahan Baku

5.2.7 Antarmuka Form Pembelian Bahan Baku

DATA BAHAN BAKU

TRANSAKSI PEMBELIAN BAHAN BAKU

Entry Data Bahan Baku

Tanggal Pembelian Jun -2012

ID Bahan Baku ☐ View

Nama Supplier

Keterangan Bahan

Harga / barang

ID Barang

Nama Barang

Jum Bahan Baku

Total Harga Rp

Tabel Transaksi Bahan Baku

ID Bahan	ID Produk	Jumlah	Jenis Bahan	Harga	Tanggal
BHN-100	MD001	1	Meter	200000	07/06/2012
BHN-103	MD001	2	Buah	8000	07/06/2012

Gambar 5.7 Antarmuka Pembelian Bahan Baku

5.2.8 Antarmuka Form Biaya Operasi

DATA BIAYA LAIN-LAIN

TRANSAKSI BIAYA OPERASIONAL

Entry Data Biaya Operasional

Bulan Operasional Jun -2012

ID Operasional

Ket Operasional

Total Biaya Operasional Rp.

Pilihan :

Simpan **Ubah** **Hapus** **Batal** **Tutup**

Tabel Data Biaya Operasional

ID BiayaOp	Keterangan	Ttl Biaya	Tanggal
Ops-100	Karyawan	4200000	03/06/2012
Ops-101	Listrik	200000	03/06/2012
Ops-102	Telpon	200000	03/06/2012
Ops-103	Air	100000	03/06/2012
Ops-104	dan lain-lain	100000	03/06/2012

Click tabel biaya operasi dua kali untuk ubah / hapus data biaya operasi

Gambar 5.8 Antarmuka Biaya Operasi

5.2.9 Antarmuka Form Data Barang baru

DATA BARANG

ENTRY DATA BARANG BARU

Entry Data Barang

Biaya Bulan Jun -2012

ID Barang

Nama Barang

Stock Awal

Keuntungan / Barang Rp.

Harga / Barang Rp.

Pilihan :







Biaya Barang Bulan, Jun-2012

Total Biaya Operasional Rp.

Biaya Operasi Per Barang Rp.

Biaya Bahan Baku Rp.

Total Biaya Produksi Rp.

Harga Pokok / Barang

Tabel Data Barang

ID Produk	Stok Awal	Keuntungan PerBarang	Harga	Tanggal
MD001	10	35000	186300	07/06/2012
MT001	10	50000	313000	07/06/2012
MT002	10	45000	308500	07/06/2012
MT003	12	50000	264167	07/06/2012

Gambar 5.9 Antarmuka Data Barang Baru

5.2.10 Antarmuka Form Stok Barang

TAMBAH STOK BARANG

DATA STOK BARANG

ENTRY DATA STOK BARANG

ID Barang ☐ View

Nama Barang

Tambah Stok

Pilihan :

 **Simpan**  **Ubah**  **Batal**  **Tutup**

TABEL STOK BARANG

	ID Produk	Stok Awal	Sisa Stok	Tanggal
▶	MD001	10	8	07/06/2012
	MT001	10	7	07/06/2012
	MT002	10	7	07/06/2012
	MT003	12	10	07/06/2012

Click tabel stok barang dua kali untuk ubah / hapus data stok barang

Gambar 5.10 Antarmuka Stok Barang

5.2.11 Antarmuka FormUpdate Diskon

UPDATE DATA DISKON

ENTRY DATA DISKON

Entry Data Barang

Barang Diatas

Mendapat Diskon (%)

Tanggal Update Mei -2012

Pilihan :

 **Update Diskon**  **Batal**  **Tutup**

Tabel Data Diskon

	Jum Barang	Persen	Tanggal	
▶	2	15	20/04/2012	

Gambar 5.11 Antarmuka Update Diskon

5.2.12 Antarmuka Form Data Supplier

DATA SUPPLIER

ENTRY DATA SUPPLIER

No Supplier

Nama Supplier

Kota

Telepon

Keterangan

Harga Barang Rp /

Pilihan :

Data Baru **Simpan** **Ubah** **Hapus** **Batal** **Tutup**

TABEL DATA SUPPLIER

ID Supl	Nama	Kota	Telepon	Keterangan		
SUP-101	wawan	Karawang	085383178502	Kulit Sapi		
SUP-102	andi	Jakarta	021-5623144	Kulit Buaya		
SUP-103	Popo	Cirebon	085722244994	Kulit ular		
SUP-104	ida herdia	Surabaya	0521448544	Benang nilon		
SUP-105	hartono	Bekasi	085222477600	Lem		
SUP-106	kartika	Bandung	022-5642133	Aksesoris dan k		
SUP-107	Sri	Bandung	08572200042	Kain Lapis		

Gambar 5.12 Antarmuka Data Supplier

5.2.13 Antarmuka Form Penjualan



TRANSAKSI PEMBELIAN BARANG

Entry Data Barang

No. Faktur: 0421130620121

ID Barang:

Nama Barang:

Harga / Barang Rp.:

Jumlah Barang:

Harga Barang Rp.: **ADD**

Barang Pilihan

No	ID Barang	Nama Barang	Harga/Barang	Jum	Harga Barang

TOTAL HARGA Rp.

TOTAL BAYAR Rp.

CASH

BAYAR

Gambar 5.13 Antarmuka Form Penjualan

5.2.14 Antarmuka Form Barang Paling Laku



The image shows a software window titled "LAPORAN BARANG PALING LAKU". Inside the window, there is a printer icon and the title "Laporan Barang Paling Laku". Below this, a section titled "Pilih berdasarakan ?" contains two radio buttons: "Perbulan" (selected) and "Pertahun". Underneath, there are two date range selectors. The first is labeled "Dari Bulan :" and the second is labeled "Sampai Bulan :". Both selectors have dropdown menus currently showing "Mei -2012". At the bottom of the window, there are two buttons: "Print View" and "Tutup".

Gambar 5.14 Antarmuka Barang Paling Laku

5.2.15 Antarmuka Form Laporan Penjualan

LAPORAN PENJUALAN

Laporan Penjualan

Pilih berdasarkan ?

☐ Perhari ☐ Perbulan ☐ Pertahun

Dari Tanggal:

09- Mei -2012

Sampai Tanggal :

09- Mei -2012

Print View **Tutup**

Gambar 5.15 Antarmuka Laporan Penjualan

5.2.16 Antarmuka Form Laporan Barang



LAPORAN BARANG

 **Laporan Barang**

Pilih berdasarkan ?

☐ ID Barang

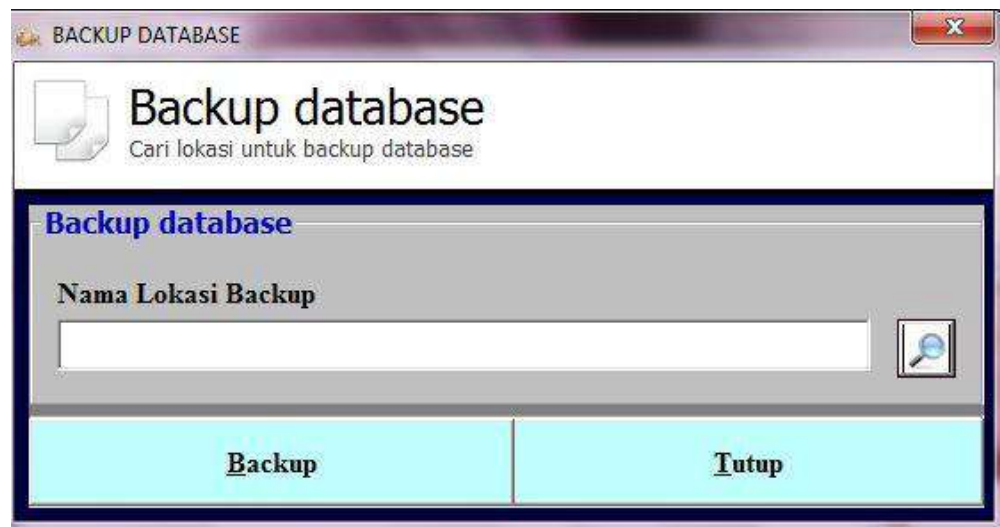
☐ Print Semua

Kata Kunci

Print View **Tutup**

Gambar 5.16 Antarmuka Laporan Barang

5.2.17 Antarmuka Form Backup Database



BACKUP DATABASE

 **Backup database**
Cari lokasi untuk backup database

Backup database

Nama Lokasi Backup



Backup **Tutup**

Gambar 5.17 Antarmuka Backup Database

5.2.18 Antarmuka Form Restore Database



RESTORE DATABASE

Restore database
Cari lokasi database yang sudah dibackup untuk melakukan restore

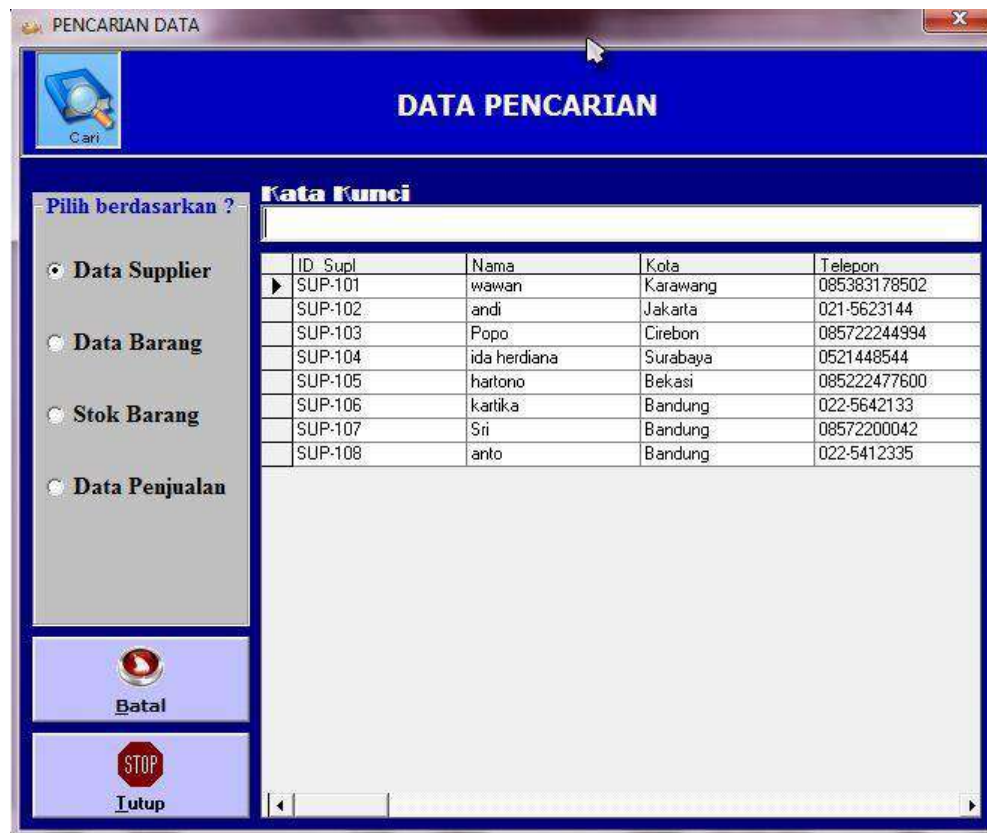
Restore database

Lokasi Database Dibackup

Restore **Tutup**

Gambar 5.18 AntarmukaRestore Database

5.2.19 Antarmuka Form View Data



PENCARIAN DATA

Cari

DATA PENCARIAN

Pilih berdasarkan ?

- ☒ Data Supplier
- ☐ Data Barang
- ☐ Stok Barang
- ☐ Data Penjualan

Kata Kunci

ID Supl	Nama	Kota	Telepon
SUP-101	wawan	Karawang	085383178502
SUP-102	andi	Jakarta	021-5623144
SUP-103	Popo	Cirebon	085722244994
SUP-104	ida herdiana	Surabaya	0521448544
SUP-105	hartono	Bekasi	085222477600
SUP-106	kartika	Bandung	022-5642133
SUP-107	Sri	Bandung	08572200042
SUP-108	anto	Bandung	022-5412335

Batal **Tutup**

Gambar 5.19 Antarmuka View Data

5.2.20 Antarmuka Form About



Gambar 5.20 Antarmuka About

5.3 Hasil Pengujian

Hasil pengujian dilakukan dengan menggunakan metode Black-Box Testing, *black-box testing* adalah metode testing perangkat lunak yang menguji fungsionalitas aplikasi yang bertentangan dengan struktur internal.

Tabel 5.2 Pengujian Perangkat Lunak

No .	Fungsi yang diuji	Cara Pengujian	Hasil Yang diharapkan	Hasil Pengujian
1	Create Account	Membuat account untuk bisa masuk ke dalam Menu Utama	Dapat login menggunakan account yang telah di buat	OK
2	Form Login	Pada Aplikasi penjualan Tas menampilkan form login untuk masuk ke	Menampilkan form login dan berhasil Masuk ke dalam menu utama	OK

		menu Utama		
3	Masukan User name dan password	Memasukan user name dan password sesuai dengan create account yang telah di buat	Bila username dan password di isi sesuai dengan yang telah di buat maka login berhasil,tetapi apabila username danpassword salah maka akan muncul messagebox	
4	Data barang	Pada Aplikasi Penjualan menampilkan data Barang	Menampilkan form data Barang yang sudah tersedia	OK
5	Data Biaya Operasional	Pada Aplikasi penjualan terdapat master data biaya operasional	Menampilkan master data pengeluaran setiap bulannya	OK
6	Data Bahan baku	Pada Aplikasi penjualan terdapat master data bahan baku	Menampilkan BahanBaku yang digunakan dalam pembuatan tas dan dompet berserta data suppliernya	
7	Pembelian bahan Baku	Memasukan bahanbaku apa saja yang di butuhkan dalam pembuatan tas	Menampilkan semua data dan mengisinya termasuk totalharga pembelian	OK
8	Biaya Operasi	Admin mengisi biaya-biaya pengeluaran dalam memproduksi	Admin bisa mengetahui biaya pengeluaran dalam pembuatan tas	OK
9	Data barang baru	Pada aplikasi penjualan dapat menambahkan stok barang	Menampilkan form data barang baru dan i isi sesuaikebutuhan	OK
10	Stok barang	Pada aplikasi penjualan terdapat stok barang yang bisa di update	Menampilkan stok barang yang telah di tambahkanmaupun stok barang yang telah terjual	OK
11	Update Diskon	Pada Aplikasi ini terdapat juga update diskon yang bisa di rubah sesuai kebutuhan	Menampilkan diskon yang telah di isi	OK
12	Entry data	Pada aplikasi	Menampilkan semua data	OK

	Supplier	penjualan terdapat form data supplier	supplier	
13	Penjualan	Pada Aplikasi penjualan terdapat transaksi pembelian barang	Mengisi dan menampilkan barang yang akan di beli kemudian menampilkan total harga yang harus dibayar	OK
14	Barang paling laku	Pada aplikasi penjualan terdapat form barang paling laku	Menampilkan barang apa saja yang paling laku terjual	OK
15	Laporan Penjualan	Pada aplikasi terdapat form laporan penjualan yang akan di pilih berdasarkan harian,bulana dan tahunan	Menampilkan laporan barang apa yang telah terjual	OK
16	Laporan Barang	Pada aplikasi terdapat form laporan barang yang akan di pilih	Menampilkan laporan barang yang terjual dan sisa barang yang masih ada.	OK
17	Backup Database	Pada aplikasi ini terdapat form backup database	Melakukan penyimpanan database	OK
18	Restore database	Pada aplikasi ini terdapat form restore database	Melakukan pengambilan database yang telah di backup	OK
19	About	Pada aplikasi penjualan terdapat menu about	Menampilkan informasi tentang Ukm	OK

Penutup

Berdasarkan hasil pengembangan sistem yang telah penulis lakukan, penulis mencoba membuat suatu kesimpulan dan mengajukan beberapa saran-saran yang berhubungan dengan pembahasan yang telah dikemukakan di bab-bab sebelumnya.

Kesimpulan

Dari berbagai penjelasan yang telah diuraikan dalam laporan ini, maka dapat disimpulkan beberapa hal sebagai berikut :

1. Aplikasi yg dibuat dapat memudahkan karyawan dalam pengolahan data barang, transaksi penjualan, laporan penjualan dan bagi pimpinan mempermudah pengambilan keputusan seperti halnya dalam pencapaian produktivitas kerja yang tinggi dan mencapai hasil yang maksimal. semuanya sudah tersimpan dalam satu tempat
2. Mempermudah proses perhitungan karena total penjualan akan muncul pada laporan.

Saran

Adapun saran yang dapat penulis berikan untuk pengembangan dari sistem yang telah dibuat antara lain :

3. Aplikasi ini dapat di kembangkan sampai kepada laporan laba rugi agar House of leather dapat mengetahui keuntungan maupun kerugian pada setiap Penjualannya.
4. Aplikasi ini dapat di kembangkan ke basis web

Source Code Menu Utama

Dim ket As String

```
Private Sub btback_Click()  
frmbackup.Show 1  
End Sub
```

```
Private Sub btback_MouseMove(Button As Integer, Shift As Integer, X As  
Single, Y As Single)lbbck.FontBold = True  
End Sub
```

```
Private Sub btbrg_Click()  
frmlapbarang.Show  
End Sub
```

```
Private Sub btbrg_MouseMove(Button As Integer, Shift As Integer, X As Single,  
Y As Single)lbbrg.FontBold = True  
End Sub
```

```
Private Sub btgrafik_Click()  
frmlaplaku.Show 1  
End Sub
```

```
Private Sub btgrafik_MouseMove(Button As Integer, Shift As Integer, X As  
Single, Y As Single)lbgra.FontBold = True  
End Sub
```

```
Private Sub btpenju_Click()  
frmtransaksi.Show 1  
End Sub
```

```
Private Sub btpenju_MouseMove(Button As Integer, Shift As Integer, X As  
Single, Y As Single)lbpen.FontBold = True  
End Sub
```

```
Private Sub btres_Click()  
frmrestore.Show 1  
End SubEnd Sub
```

```
Private Sub lgin_MouseMove(Button As Integer, Shift As Integer, X As Single,  
Y As Single)lblg.FontBold = True  
End Sub
```

```
Private Sub lgo_Click()lgout  
MsgBox "Program sudah di logout", vbInformation,  
"Information"End Sub  
Sub lgout()nonaktif
```

```
StatusBar1.Panels(2) = ""
StatusBar1.Panels(4) = "" End
Sub
Private Sub MDIForm_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)lbbold
End Sub

Private Sub MDIForm_Unload(Cancel As Integer)
MsgBox "Terimakasih sudah menggunakan program ini", vbInformation,
"Information" End Sub

Private Sub cf_Click()
frmDataBarang.Show End
Sub

Private Sub dd_Click()
frmstok.Show
End Sub
```

```
Private Sub ds_Click()  
frmlaptran.Show 1 End Sub
```

```
Private Sub FEDS_Click()  
frmdtsupl.Show 1  
End Sub
```

```
Private Sub ff_Click()  
frmdis.Show 1  
End Sub
```

```
Private Sub FSCH_Click()  
frmcaridata.Show 1  
End Sub
```

```
Private Sub kelu_Click()  
Dim tanya  
tanya = MsgBox("Apakah Anda Yakin Akan Keluar Dari Aplikasi Ini?", vbQuestion + vbYesNo, "Keluar")
```

```
If tanya = vbYes Then End  
Else 'Reset End If  
End Sub
```

```
Private Sub klr_Click()  
Dim tanya  
tanya = MsgBox("Apakah Anda Yakin Akan Keluar Dari Aplikasi Ini?", vbQuestion + vbYesNo, "Keluar")
```

```
If tanya = vbYes Then End  
Else End If End Sub
```

```
Private Sub Lg_Click()  
frmlogin.Show 1  
End Sub
```

```
Private Sub lgin_Click()  
frmlogin.Timer1 = True End  
Sub
```

```
Sub lbbold() lblg.FontBold =  
False lbsup.FontBold = False  
lbbrg.FontBold = False  
lbpen.FontBold = False  
lbgra.FontBold = False  
lbbck.FontBold = False  
lbres.FontBold = False  
lbvie.FontBold = False  
lbkel.FontBold = False End  
Sub
```

```
Private Sub FCU_Click()  
datauser.Show 1
```

```
End Sub
Private Sub inf_Click()
frminfo.Show 1
End Sub
Private Sub winCascade_Click()
Me.Arrange vbCascade
End Sub

Private Sub winHori_Click()
Me.Arrange vbTileHorizontalEnd
Sub
```

```

Private Sub winVert_Click()
Me.Arrange vbTileVertical End
Sub
Private Sub MDIForm_Activate()
N = 1
k = 1
ket = "^_^" & Label1.Caption
Timer1.Enabled = True
StatusBar1.Panels(2) = nmusr
StatusBar1.Panels(4) = Level End
Sub

```

```

Private Sub MDIForm_Load()
nonaktif
StatusBar1.Panels(6) = Format(Date, "dd-MMM-
yyyy")End Sub

```

```

Private Sub nmpro_Click()
frmproduk.Show 1
End Sub

```

```

Private Sub Picture1_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)lbbold
End Sub

```

```

Private Sub Picture2_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)lbbold
End Sub

```

```

Private Sub Timer1_Timer()
ket = Right(ket, Len(ket) - 1) & Left(ket, 1)
Label1.Caption = ket
StatusBar1.Panels(8) = Format(Time, "hh:mm:ss")
End Sub

```

```

Sub nonaktif() lgo.Enabled =
False FCU.Enabled = False
dt.Enabled = False ds.Enabled
= False btbrg.Enabled = False
btpenju.Enabled = False
btgrafik.Enabled = False
btback.Enabled = False
btres.Enabled = False
btview.Enabled = False
lgin.Enabled = True
lg.Enabled = True

```

```

lblg.Enabled = True
lbsup.Enabled = False
lbbrg.Enabled = False

```

```
lbpen.Enabled = False
lbgra.Enabled = False
lbbck.Enabled = False
lbres.Enabled = False
lbvie.Enabled = False End
Sub
```

```
Private Sub tr_Click()
frmbiaya.Show 1
End Sub
```

```
Private Sub uu_Click()
frmbhnbaku.Show 1 End Sub
```

Source Code login

Option Explicit

```
Const LWA_BOTH = 3 Const
LWA_ALPHA = 2
Const LWA_COLORKEY = 1 Const
GWL_EXSTYLE = -20
Const WS_EX_LAYERED = &H80000
```

```
Private Declare Function GetWindowLong Lib "user32" Alias _
"GetWindowLongA" (ByVal hwnd As Long, ByVal nIndex As Long) As
Long
Private Declare Function SetWindowLong Lib "user32" Alias _
"SetWindowLongA" (ByVal hwnd As Long, ByVal nIndex As Long,
ByVal dwNewLong As Long) As Long
Private Declare Function SetLayeredWindowAttributes Lib "user32" _
(ByVal hwnd As Long, ByVal color As Long, ByVal X As Byte, _
ByVal alpha As Long) As Boolean
Dim TransparanDonk As Integer
Sub TransparanBro(hwndBro As Long, TransBro As
Integer) On Error Resume Next
```

```
Dim OKBro As Long
OKBro = GetWindowLong(hwndBro, GWL_EXSTYLE)
```

```
SetWindowLong hwndBro, GWL_EXSTYLE, OKBro Or
WS_EX_LAYERED SetLayeredWindowAttributes hwndBro, RGB(255,
255, 0), TransBro, LWA_ALPHA Exit Sub
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
TransparanDonk = 0 hilang
End Sub
```

```
Private Sub Label1_Click() End
```

Sub

```
Private Sub Timer1_Timer() On
Error Resume Next
TransparanDonk = TransparanDonk + 5
If TransparanDonk > 255 Then TransparanDonk = 255:
Timer1.Enabled = False TransparanBro Me.hwnd, TransparanDonk
Me.Show End Sub
Private Sub cmdcancel_Click()
TransparanDonk = 0 hilang
End Sub Sub hilang()
Dim cepat cepat = 30
While Left + Width < Screen.Width
```

```
DoEvents
Left = Left + cepatWend
While Top - Height < Screen.Height
DoEvents
Top = Top + cepatWend
Unload MeEnd Sub
Private Sub cmdOK_Click()
masuk
End Sub

Private Sub Form_Load()Main
Call Linkusr
```

```

End Sub Sub masuk()
If txuser.Text = "" Then
MsgBox "ID user belum diisi", vbInformation,
"Information"txuser.SetFocus
Exit Sub
End If
If txpass.Text = "" Then
MsgBox "Password belum diisi", vbInformation,
"Information"txpass.SetFocus
Exit SubEnd If

```

```

If txpass = pas And Level = "Admin" Then
MDIForm1.lgo.Enabled = True
MDIForm1.FCU.Enabled = True
MDIForm1.dt.Enabled = True
MDIForm1.ds.Enabled = True
MDIForm1.btbrg.Enabled = True
MDIForm1.btpenju.Enabled = True
MDIForm1.btgrafik.Enabled = True
MDIForm1.btback.Enabled = True
MDIForm1.btres.Enabled = True
MDIForm1.btview.Enabled = True
MDIForm1.lgin.Enabled = False
MDIForm1.lg.Enabled = False

```

```

MDIForm1.lblg.Enabled = False
MDIForm1.lbsup.Enabled = True
MDIForm1.lbbrg.Enabled = True
MDIForm1.lbpen.Enabled = True
MDIForm1.lbgra.Enabled = True
MDIForm1.lbbck.Enabled = True
MDIForm1.lbres.Enabled = True
MDIForm1.lbvie.Enabled = True
hilang
Exit SubEnd If

```

```

If txpass = pas And Level = "Operator" Then
MDIForm1.lgo.Enabled = True
MDIForm1.FCU.Enabled = False
MDIForm1.dt.Enabled = False
MDIForm1.ds.Enabled = True
MDIForm1.btbrg.Enabled = True
MDIForm1.btpenju.Enabled = True
MDIForm1.btgrafik.Enabled = True
MDIForm1.btback.Enabled = True
MDIForm1.btres.Enabled = True
MDIForm1.btview.Enabled = True
MDIForm1.lgin.Enabled = False

```

MDIForm1.lg.Enabled = False

MDIForm1.lblg.Enabled = False

MDIForm1.lbsup.Enabled = True

MDIForm1.lbbrg.Enabled = True

MDIForm1.lbpen.Enabled = True

MDIForm1.lbgra.Enabled = True

MDIForm1.lbbck.Enabled = True

MDIForm1.lbres.Enabled = True

MDIForm1.lbvie.Enabled = True

hilang

Exit Sub End If End

If

MsgBox "ID user tidak ditemukan", vbInformation,

"Information" txuser.SetFocus: txuser = ""

End With

lanjut:
End Sub

Private Sub btlog_Click() masuk
End Sub

Private Sub tpass_Change()
tpass.PasswordChar = "*" End
Sub

Private Sub tpass_KeyDown(KeyCode As Integer, Shift As
Integer) If KeyCode = 13 Then
If tpass = "" Then
MsgBox "Password belum diisi", vbInformation, "Information"
tpass.SetFocus
Exit Sub End If
masuk End If
End Sub

Private Sub txuser_Change()
txuser = StrConv(txuser, 1)
txuser.SelStart = Len(txuser) End
Sub

Private Sub txuser_KeyDown(KeyCode As Integer, Shift As
Integer) If KeyCode = 13 Then
If txuser = "" Then
MsgBox "ID User belum diisi", vbInformation, "Information"
txuser.SetFocus
Exit Sub End If
tpass.SetFocus
End If End Sub

SourceCode pembelian Bahan Baku

Dim idsuppl, kdtes, tgl, kodnot, nmpro, jns1, hrg1

Private Sub cmdcancel_Click()
bersih
End Sub Sub bersih()
Call Form_Load idbhn =
"" idsuppl = "" kdtes = ""
nmbhn = ""
hrg = "" jumbhn = ""
ttlbn = "" nmsupp =
"" ket = ""
jns = "" nmbrg = ""
vi.Value = 0
vi.Enabled = True idbhn.Enabled
= True cmdsave.Enabled = False

```
cmdedit.Enabled = False
cmdcancel.Enabled = False
cmddelete.Enabled = False
Frame4.Enabled = False
DTPicker1.Enabled = True
idbhn.SetFocus
End Sub
```

```

Private Sub cmddelete_Click()
If MsgBox("Yakin data bahan baku ingin dihapus...?", vbYesNo + vbInformation, "Konfirmasi") =
vbYes Then sambung.Execute "Delete From DataBhnBaku where ID_Bahan=" & idbhn & "and
ID_Produk=" & nmbhn & "and
Month(Tanggal)=" & Month(DTPicker1) & "and Year(Tanggal)=" &
Year(DTPicker1) & "" MsgBox "Data Sudah Dihapus", vbInformation,
"Sukses"
bersih
Else End If End Sub

```

```

Sub ubah()
If nmbrg = "" Then
MsgBox "ID barang belum diisi", vbInformation,
"Information" nmbhn.SetFocus
Exit Sub End If

```

```

If MsgBox("Yakin Data Bahan Baku Diubah,,,,?", vbYesNo + vbInformation, "Konfirmasi") =
vbYes Then sambung.Execute "UPDATE DataBhnBaku SET ID_Produk=" & nmbhn &
",Jumlah=" & jumbhn & ",Harga=" & _ttlbn & "where ID_Bahan=" & idbhn & "and
ID_Produk=" & nmbhn & "and Month(Tanggal)=" &
Month(DTPicker1) & "and Year(Tanggal)=" &
Year(DTPicker1) & "" MsgBox "Data Sudah Diubah",
vbInformation, "Information" bersih
End If End Sub

```

```

Private Sub cmdedit_Click() ubah
End Sub

```

```

Private Sub cmdexit_Click()
Unload Me
End Sub

```

```

Private Sub DataGrid1_DblClick()
kodnot = "1"
With TBLBhnbaku
Frame4.Enabled = True
idbhn = TBLBhnbaku!ID_Bahan
nmbhn = TBLBhnbaku!ID_Produk
jumbhn = TBLBhnbaku!Jumlah ttlbn
= TBLBhnbaku!Harga DTPicker1 =
TBLBhnbaku!Tanggal End With
kodnot = ""
If TBLMasBaku.State = 1 Then TBLMasBaku.Close
TBLMasBaku.Open "SELECT * FROM DataMasBaku WHERE ID_BahanBaku like%" & idbhn & "%""
With TBLMasBaku
idsupl = TBLMasBaku!ID_Supl End
With

```

```

If TBLSupl.State = 1 Then TBLSupl.Close
TBLSupl.Open "SELECT * FROM DataSupplier WHERE ID_Supl like%" & idsupl & "%""

```

```
With TBLSuppl
nmsupp = TBLSuppl!Nama ket =
TBLSuppl!Keterangan hrg1 =
TBLSuppl!Harga jns1 =
TBLSuppl!Jenis
hrp = " " + Format(hrg1, "##,##")jns
= "Per" + " " + jns1
End With
```

```
If TBLProduk.State = 1 Then TBLProduk.Close
TBLProduk.Open "SELECT * FROM DataProduk WHERE ID_Produk like%" & nmbhn & "%"
With TBLProduk
nmbrg = TBLProduk!Nama End
With
```

```
idbhn.Enabled = False
```

```

If TBLSuppl.State = 1 Then TBLSuppl.Close
TBLSuppl.Open "SELECT * FROM DataSuplier WHERE ID_Supl like'" &
idsuppl & "'""nmsupp = TBLSuppl!Nama
ket = TBLSuppl!Keterangan hrg1
= TBLSuppl!Harga jns1 =
TBLSuppl!Jenis
hrg = " " + Format(hrg1, "##,##")jns
= "Per" + " " + jns1
Frame4.Enabled = True
kdtes = "" vi.Value = 0
nmbhn.SetFocus End Sub
Private Sub DTPicker1_Change()
If Year(DTPicker1) >= Year(tgl) And Month(DTPicker1) > Month(tgl) Then
MsgBox "Bulan pembelian bahan baku tidak boleh lebih dari bulan sekarang",
vbInformation, "Information" Call Form_Load
Exit Sub End If
idbhn.SetFocus carihbn
End Sub
Private Sub Form_Load() Main
Call Linkbhnbaku Call
Linksuppl Call Linkproduk
Call Linkmasbaku
Set DataGrid1.DataSource = TBLBhnbaku
Set DataGrid2.DataSource = TBLMasBaku
DTPicker1 = Date
tgl = Date eneb
tampro carihbn End
Sub
Sub tampro()
nmbhn.Clear
For a = 1 To TBLProduk.RecordCount
nmbhn.AddItem TBLProduk!ID_Produk
TBLProduk.MoveNext
Next a End Sub
Sub carihbn()
If TBLBhnbaku.State = 1 Then TBLBhnbaku.Close
TBLBhnbaku.Open "SELECT * FROM DataBhnBaku WHERE Month(Tanggal) like'" &
Month(DTPicker1) & "'and Year(Tanggal) like'" & Year(DTPicker1) & "'""
With DataGrid1
.ReBind End With
End Sub Sub eneb()
DataGrid2.Columns(1).Visible = False
End Sub
Sub aktif() Frame1.Enabled =
True cmdcancel.Enabled = True
cmdsave.Enabled = True
cmddelete.Enabled = False
cmdedit.Enabled = False
Frame3.Enabled = True End

```

Sub

Private Sub Frame3_DragDrop(Source As Control, X As Single, Y

As Single)End Sub

*Private Sub idbhn_KeyDown(KeyCode As Integer, Shift As
Integer)If KeyCode = 13 Then*

```

If idbhn = "" Then
MsgBox "ID bahan belum dicari", vbInformation,
"Information" idbhn.SetFocus
Exit Sub End If
pindahsupp
End If End Sub

```

```

Private Sub jumbhn_Change()
If jumbhn = "" Then ttlbhn =
""
Else
ttlbhn = Val(jumbhn) * Val(hrg1)
ttlbhn = Format(ttlbhn, "##,##") End
If
End Sub

```

```

Private Sub jumbhn_KeyPress(KeyAscii As
Integer) KeyAscii = Asc(UCase(Chr(KeyAscii)))
If KeyAscii = 13 Then End If
If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack) Then Beep
KeyAscii = 0
'MsgBox "Hanya Angka Bisa Diisi", vbInformation,
"Information" End If
End Sub

```

```

Private Sub nmbhn_Change() If
nmbhn = "" Then
Else
If kodnot = "" Then kodnot
= ""
MsgBox "Pilih ID barang", vbInformation, "Information"
nmbhn = ""
nmbrg = ""
Else
kodnot = "1" End If
End If End Sub

```

```

Private Sub nmbhn_Click()
caripro
End Sub
Sub caripro()
On Error GoTo akodnot
= "1"
If TBLProduk.State = 1 Then TBLProduk.Close
TBLProduk.Open "SELECT * FROM DataProduk WHERE ID_Produk like '%" &
nmbhn & "%'" nmbhn = TBLProduk!ID_Produk
nmbrg = TBLProduk!Nama kodnot
= ""

```

```
jumbhn.SetFocus
If idbhn.Enabled = False ThenElse
cmdsave.Enabled = True
cmdcancel.Enabled = TrueEnd
If
a:
End Sub
Private Sub nmbhn_KeyPress(KeyAscii As
Integer)nmbhn = StrConv(nmbhn, 3)
nmbhn.SelStart = Len(nmbhn)End
Sub
Private Sub idbhn_Change()
carisup
End Sub
```

```

Sub carisup()
If idbhn = "" Then vi.Value
= 0
Else vi.Value = 1
If kdtes = "" Then
If TBLMasBaku.State = 1 Then TBLMasBaku.Close
TBLMasBaku.Open "SELECT * FROM DataMasBaku WHERE ID_BahanBaku like '%" &
idbhn & "%'" With DataGrid2
.ReBind End With eneb
If TBLMasBaku.RecordCount = 0 Then
MsgBox "Data bahan baku tidak ditemukan", vbInformation,
"Information" idbhn = ""
Exit Sub End If
End If End If End
Sub
Private Sub vi_Click() If
vi.Value = 1 Then
DataGrid2.Visible = True Else
DataGrid2.Visible = False End
If
End Sub
Private Sub cmdsave_Click()
simpan
End Sub
Sub simpan()
If idbhn = "" Then
MsgBox "ID bahan baku belum dipilih", vbInformation,
"Information" idbhn.SetFocus
Exit Sub End If

If nmbrg = "" Then
MsgBox "ID barang belum diisi", vbInformation,
"Information" nmbhn.SetFocus
Exit Sub End If

sambung.Execute "Insert into
DataBhnBaku(ID_Bahan,ID_Produk,Jumlah,Jenis_Bahan,Harga,Tanggal)VALUES('" & idbhn &
-
"', '" & nmbhn & "', '" & jumbhn & "', '" & jns1 & "', '" & ttlbhn & "', '" &
DTPicker1 & "')" bersih
MsgBox "Data sudah disimpan", vbInformation, "Information"

```

End Sub **Source Code Transaksi Penjualan**

```

Dim kodthn, kodcari, kodhr, hrg1, krghrg, koddis
Dim ttlhr, kodjam, kodmnt, kodhr, kodbln, jam, tgl, smtg1,
smtgl Sub pinbrg()
kodcari = "1"
With TBLBarang
idbrg = TBLBarang!ID_Produk

```

```
kodhrng = TBLBarang!Harga  
hrnbrg = Format(TBLBarang!Harga, "###,###")  
End With
```

```
If TBLProduk.State = 1 Then TBLProduk.Close  
TBLProduk.Open "SELECT * FROM DataProduk WHERE ID_Produk like'" &  
idbrg & "%'"nmbrg = TBLProduk!Nama  
DataGrid1.Visible = False  
kodcari = ""  
jum.SetFocus cmdcancel.Enabled  
= True  
End Sub
```

```

Private Sub cmdbyr_Click()
If kodjum = "" Or kodjum = "0" Then
MsgBox "Data barang belum ditentukan", vbInformation,
"Information" idbrg.SetFocus
Exit Sub
End If

```

```

If ttlchas = "" Or ttlchas = "0" Then
MsgBox "Uang cash belum diisi", vbInformation,
"Information" ttlchas.SetFocus
Exit Sub End If

```

```

If kem < 0 Then
MsgBox "Uang cash tidak mencukupi, silahkan tambah lagi", vbInformation,
"Information" ttlchas.SetFocus
Exit Sub End If
simpan End Sub
Sub simpan() sambung.Execute
"Insert into
DataPenjualan(ID_Faktur,Ttl_Item,Ttl_Harga,Persen,Ttl_Bayar,Ttl_Chass,Kembalian,Tanggal,Nama
User)VALUES('' & nof & _
'', '' & kodjum & '', '' & ttlhr & '', '' & disk & '', '' & ttlbyr & '', '' & ttlchas & '', '' & kem &
'', '' & tgl & '', '' & nmusr & '')"

```

```

For e = 0 To List1.ListCount - 1
If TBLStok.State = 1 Then TBLStok.Close
TBLStok.Open "SELECT * FROM DataStok WHERE ID_Produk like%" & List2.List(e) & "%"
sambung.Execute "UPDATE DataStok SET Sisa_Stok=" & Val(TBLStok!Sisa_Stok) -
Val(List5.List(e)) & "where ID_Produk=" & List2.List(e) & ""
sambung.Execute "Insert into DetailPenjualan
(ID_Faktur,ID_Produk,Jum_Barang,Jum_Harga)VALUES('' & nof & '', '' & List2.List(e) & '', ''
& List5.List(e) & '', '' & List6.List(e) & '')"
Next e

```

```

kembali.nof = nof
kembali.beres = kem
kembali.Show 1 bersih
End Sub

```

```

Sub bersih() Call
Form_Load idbrg = ""
nmbrg = "" hrgrg = ""
jum = ""
hrgrg = "" hrgr1 = ""
ttlhrgr = "" ttlhrgr1 = ""
ttlbyr = "" ttlchas = ""
kem = "" disk = ""
ketdis = "" kodno = ""
kodjum = "" kodstk =

```

'''

idbrg.SetFocus

List1.Clear List2.Clear

List3.Clear List4.Clear

List5.Clear List6.Clear

End Sub

```
Private Sub cmdcancel_Click()  
bersih  
End Sub
```

```
Private Sub cmdexit_Click()  
Unload Me  
End Sub
```

```
Private Sub DataGrid1_DblClick()  
pinbrg  
End Sub
```

```
Private Sub Form_Activate()  
idbrg.SetFocus  
End Sub
```

```
Private Sub Form_Load()Main  
Call LinkbarangCall  
Linkstok Call Linkproduk  
Call Linkpenju  
Call Linkdetailpenju Call  
Linkdis  
Set DataGrid1.DataSource = TBLBarang  
tgl = Format(Date, "dd-MMM-yyyy")  
smtgl = Format(Date, "MM")  
jam = Format(Time, "hh:mm:ss")pot  
hilang databaru End  
Sub Sub hilang()  
DataGrid1.Columns(1).Visible = False  
DataGrid1.Columns(2).Visible = False  
DataGrid1.Columns(3).Visible = False  
DataGrid1.Columns(4).Visible = False  
End Sub  
Sub databaru()  
Dim potkk, angka, kosong, potnm, pjg As String  
If TBLPenju.State = 1 Then TBLPenju.Close  
TBLPenju.Open "select*from DataPenjualan",  
sambungamatanggal  
If TBLPenju.EOF = False Then  
TBLPenju.MoveFirst  
Do  
potkk = TBLPenju(0)  
TBLPenju.MoveNext Loop Until  
TBLPenju.EOF  
potkk = Trim(Mid(potkk, 13, 9))angka  
= Val(potkk) + 1  
pjg = Len(angka)  
For kode = 1 To 13 - pjgNext
```

```
kode  
kosong = Right((angka), 9)  
nof.Text = kodmnt + kodjam + kodthn + kodbln + kodhr +  
kosong Else  
nof.Text = kodmnt + kodjam + kodthn + kodbln + kodhr +  
"100" End If
```

```
If TBLPenju.RecordCount <> 0 Then  
TBLPenju.MoveFirst  
End If End Sub Sub  
pot()  
kodjam = Trim(Mid(jam, 1, 2))  
kodmnt = Trim(Mid(jam, 4, 2))  
kodthn = Format(Date, "yyyy")
```

```

kodbln = Format(Date, "MM")
kodhr = Format(Date, "dd") End
Sub
Sub caribrg() hilang
If idbrg = "" Then
DataGrid1.Visible = False nmbrg
= ""
hrgbrg = ""
jum = ""
hrg = "" Else
DataGrid1.Visible = True
If TBLBarang.State = 1 Then TBLBarang.Close
TBLBarang.Open "SELECT * FROM DataBarang WHERE ID_Produk like '%" &
idbrg & "%'" With DataGrid1
.ReBind End With
If TBLBarang.RecordCount = 0 Then
MsgBox "Data barang tidak ditemukan", vbInformation,
"Information" idbrg = ""
hilang Exit Sub End If
End If End Sub

```

```

Private Sub Frame1_DragDrop(Source As Control, X As Single, Y

```

```

As Single) End Sub

```

```

Private Sub idbrg_Change()
If kodcari = "" Then caribrg
Else End If
idbrg = StrConv(idbrg, 1)
idbrg.SelStart = Len(idbrg)
hilang
End Sub

```

```

Private Sub idbrg_KeyDown(KeyCode As Integer, Shift As
Integer) If KeyCode = 13 Then
If idbrg = "" Then
MsgBox "Kode Barang belum diisi", vbInformation,
"Information" idbrg.SetFocus
Exit Sub End If
pinbrg
End If End Sub

```

```

Private Sub jum_Change()
If jum = "" Then hrg = ""
tam.Enabled = False Else
hrg1 = Val(jum) * Val(kodhr) hrg
= Format(hrg1, "###,###")
tam.Enabled = True
End If End Sub

```

```
Private Sub jum_KeyDown(KeyCode As Integer, Shift As  
Integer) If KeyCode = 13 Then  
If jum = "" Then  
MsgBox "Jumlah Barang belum diisi", vbInformation,  
"Information"jum.SetFocus  
Exit Sub
```

```
End Ifkelist
    End If
End Sub
```

```
Private Sub jum_KeyPress(KeyAscii As Integer)
KeyAscii = Asc(UCase(Chr(KeyAscii)))
If KeyAscii = 13 Then End If
If Not (KeyAscii >= Asc("0") And KeyAscii <= Asc("9") Or KeyAscii =
vbKeyBack) Then Beep
KeyAscii = 0
'MsgBox "Hanya Angka Bisa Diisi", vbInformation,
"Information" End If
End Sub
```

```
Private Sub List1_Click()
List6.ListIndex = List1.ListIndex
List2.ListIndex = List1.ListIndex
List3.ListIndex = List1.ListIndex
List4.ListIndex = List1.ListIndex
List5.ListIndex = List1.ListIndex End
Sub
```

```
Private Sub List1_DblClick()
dataremov
End Sub
```

```
Private Sub List2_Click()
List1.ListIndex = List2.ListIndex
List6.ListIndex = List2.ListIndex
List3.ListIndex = List2.ListIndex
List4.ListIndex = List2.ListIndex
List5.ListIndex = List2.ListIndex End
Sub
```

```
Private Sub List2_DblClick()
dataremov
End Sub
```

```
Private Sub List3_Click()
List1.ListIndex = List3.ListIndex
List2.ListIndex = List3.ListIndex
List6.ListIndex = List3.ListIndex
List4.ListIndex = List3.ListIndex
List5.ListIndex = List3.ListIndex End
Sub
```

```
Private Sub List3_DblClick()
dataremov
End Sub
```

```
Private Sub List4_Click()  
List1.ListIndex = List4.ListIndex  
List2.ListIndex = List4.ListIndex  
List3.ListIndex = List4.ListIndex  
List6.ListIndex = List4.ListIndex  
List5.ListIndex = List4.ListIndex  
End Sub
```

```
Private Sub List4_DblClick()  
dataremov  
End Sub
```

```
Private Sub List5_Click()  
List1.ListIndex = List5.ListIndex  
List2.ListIndex = List5.ListIndex  
List3.ListIndex = List5.ListIndex  
List4.ListIndex = List5.ListIndex
```

```
List6.ListIndex = List5.ListIndex End  
Sub
```

```
Private Sub List5_DblClick()  
dataremov  
End Sub
```

```
Private Sub List6_Click()  
List1.ListIndex = List6.ListIndex  
List2.ListIndex = List6.ListIndex  
List3.ListIndex = List6.ListIndex  
List4.ListIndex = List6.ListIndex  
List5.ListIndex = List6.ListIndex End  
Sub  
Sub dataremov()  
kodno = Val(kodno) - 1  
kodjum = Val(kodjum) - Val(List5.Text)  
krghrg = Format(List6.Text, "#####")  
ttlhrq1 = Val(Format(ttlhrq1, "#####")) -  
Val(krghrg)ttlhrq = Format(ttlhrq1, "###,###")  
hitdisk  
List2.RemoveItem (List2.ListIndex)  
List3.RemoveItem (List3.ListIndex)  
List4.RemoveItem (List4.ListIndex)  
List5.RemoveItem (List5.ListIndex)  
List6.RemoveItem (List6.ListIndex)  
List1.Clear  
For r = 0 To List2.ListCount - 1  
List1.AddItem r + 1  
Next r idbrg.SetFocus  
End Sub
```

```
Private Sub List6_DblClick()  
dataremov  
End Sub  
Sub htqstok()  
For cek = 0 To List2.ListCount - 1  
If DataGrid1.Columns(0) = List2.List(cek) Then  
MsgBox "Data barang sudah tersedia dilist barang pilihan", vbInformation,  
"Information" idbrg.SetFocus  
idbrg = "" Exit Sub  
End If Next cek
```

```
If TBLStok.State = 1 Then TBLStok.Close  
TBLStok.Open "SELECT * FROM DataStok WHERE ID_Produk like%" &  
idbrg & "%"" If TBLStok.RecordCount = 0 Then  
Exit Sub End If
```

```
With TBLStok  
kodstk = TBLStok!Sisa_Stok If
```

```
Val(kodstk) <= 0 Then
MsgBox "Stok barang sudah habis", vbInformation,
"Information" idbrg.SetFocus
idbrg = "" Else
If Val(jum) > Val(kodstk) Then
MsgBox "Stok barang tidak mencukupi, Sisa Stok = " + kodstk, vbInformation,
"Information" jum.SetFocus
Exit Sub
End If dtstk End If
End With
Call Form_Load End Sub
```

```

Sub dtstk()
kodno = Val(kodno) + 1
kodjum = Val(kodjum)
+ Val(jum)ttlhr1 =
Val(ttlhr1) +
Val(hrg1) ttlhr =
Format(ttlhr1,
"###,###")hitdisk
List1.AddItem
kodno
List2.AddItem
idbrg
List3.AddItem
nmbrg
List4.AddItem
hrgrg
List5.AddItem
jum
List6.AddItem
hrgr

```

```

idbrg
= ""
nmbrg
= ""
hrgrg
= ""
jum =
""
hrgr = ""
idbrg.Set
Focus
End Sub

```

```

Private Sub
tam_Click()
kelist
End
Sub
Sub
kelist()
If nmbrg = "" Then
MsgBox "Data barang belum ditentukan",
vbInformation, "Information" idbrg.SetFocus
jum
= ""
Exit
Sub
End
If

```

htgst

ok

End

Sub