BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Dalam menentukan objek penelitian, penulis melakukan penelitian pada Butik Yani Collection yang beralamat di Jl.Toha Ramdahan No.80 Kab.Bandung.

3.1.1 Sejarah Singkat Perusahaan

Butik Yani Collection adalah sebuah butik yang menyediakan pakaian busana muslim dengan mengusung "Spiritualitas Islam yang Mencerahkan" melalui kekuatan warna,variasi motif, design simple dan matching yang terdiri atas paduan kain kaos terpilih dan kain katun berkualitas. Butik Yani Collection merupakan salah satu butik yang ternama diKabupaten Bandung, Berangkat dari sebuah usaha kecil rumahan dan kios kecil-kecilan dari tahun 2006 kini terus bekembang hingga sekarang. Tujuan didirikannya butik ini adalah membangun usaha butik diKabupaten Bandung ini semakin maju. Hingga saat ini Butik Yani Collection telah melayani pengiriman ke luar kota seperti sumedang,tasikmalaya,banjar, dan ciamis.

3.1.2 Visi dan Misi Perusahaan

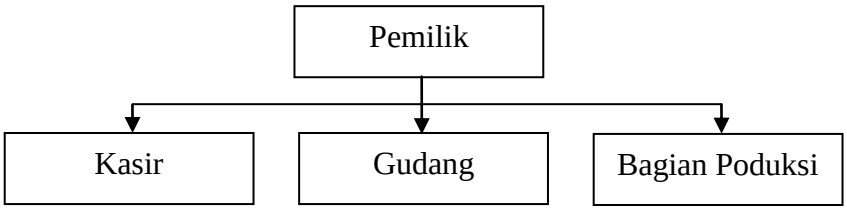
Visi butik adalah memperluas dan mempertahankan usaha yang sedang di jalankan, serta memberikan pelayanan baik pada pelanggan.

Sedangkan Misi butik adalah “Spiritualitas Islam yang Mencerahkan” melalui kekuatan warna,variasi motif, design simple dan matching yang terdiri atas paduan kain kaos terpilih dan kain katun berkualitas.

3.1.3 Struktur Organinsasi Perusahaan

Struktur organisasi merupakan salah satu komponen penting dalam pembentukan sebuah perusahaan, agar dapat mampu bersaing dan mencapai tujuan perusahaan yang diinginkan. Melalui sistem organisasi tersebut akan diatur tugas dan wewenang kepada masing-masing personil perusahaan. Tanpa adanya struktur organisasi yang baik, tidak mungkin perusahaan akan bekerja baik sesuai dengan apa yang diinginkan.

Ini berarti bahwa struktur organisasi akan sangat berperan penting terhadap kelangsungan hidup perusahaan. Struktur organisasi Butik Yani Collection dapat digambarkan sebagai berikut:



Gambar 3.1 Struktur Organisasi

3.1.4 Deskripsi Tugas

Berdasarkan susunan struktur di atas dapat kita jelaskan tentang kewajiban maupun wewenang dari masing-masing Bagian yaitu :

1. Pemilik

Adalah pemilik yang mempunyai hak untuk membuat keputusan di dalam

2. Kasir

   Mempunyai tugas dan melayani konsumen di bidang transaksi keuangan dalam penjualan barang.

3. Gudang

   Mempunyai tugas di gudang untuk mengecek barang dan mengatur segala macam aktifitas gudang.

4. Bagian Produksi

   Mempunyai tugas untuk memproduksi barang, apabila stok barang di butik sudah mencapai pada stok minimum atau habis.

3.2 Metode Penelitian

Suatu penelitian tidak akan berjalan dengan baik apabila tidak dilakukan dengan metode-metode yang baik untuk mendukung penelitian. Metode yang dilakukan penulis yaitu metode *Action*. Metode ini merancang dan mendesain program berdasarkan analisis sistem yang diusulkan sesuai dengan masalah yang ada di dunia lapangan.

3.2.1 Jenis dan Metode Pengumpulan Data

Untuk memperoleh data yang diperlukan, penulis menggunakan metode sebagai berikut :

3.2.1.1 Sumber Data Primer

Data primer yaitu data yang diperoleh langsung dari sumber tempat yang diteliti yaitu di Butik Yani Collection . Untuk mendapatkan data primer, dilakukan hal sebagai berikut :

a. Observasi, yaitu teknik pengumpulan data dan informasi yang dilakukan dengan cara pengamatan langsung terhadap tempat penelitian. Sehingga penulis dapat mengetahui alur sistem yang dilakukan pada Butik Yani Collection.
b. Wawancara, yaitu teknik pengumpulan data dengan cara tanya jawab langsung dengan para pegawai dan pemilik Butik Yani Collection. Hal ini bertujuan untuk mendapatkan data serta informasi yang benar dan akurat dengan masalah yang diteliti penulis. Sehingga penulis dapat mengetahui sistem yang sedang berjalan di Butik Yani Collection.

3.2.1.2 Sumber Data Sekunder

Cara yang digunakan untuk mengumpulkan data sekunder adalah metode Dokumentasi, yaitu dengan mengumpulkan data dan informasi yang diperlukan dari sumber-sumber kebanyakan dari materi sejenis dokumen yang berkenaan dengan masalah yang diteliti. Faktur penjualan merupakan dokumen yang digunakan pada sistem penjualan di Butik Yani Collection saat ini. Metode ini digunakan untuk pengumpulan data yang berhubungan dengan sejarah, tujuan, kegiatan dan struktur organisasi.

3.2.2 Metode Pendekatan dan Pengembangan Sistem

Adapun metode pendekatan dan pengembangan sistem yang digunakan oleh penulis adalah sebagai berikut :

3.2.2.1 Metode Pendekatan Sistem

Metode yang digunakan adalah dengan metode pendekatan objek . Metode berorientasi objek diperkenalkan pada tahun 1980, menggunakan perangkat kerja dan teknik-teknik yang dibutuhkan dalam pengembangan sistem, yaitu dynamic dan *static object oriented model*, *state transition diagram* dan *case scenario*.

Fokus utama metode pada objek, dengan melihat suatu sistem terdiri dari objek yang saling berhubungan. Objek dapat digambarkan sebagai benda, orang, tempat dan sebagainya yang mempunyai atribut dan metode. Metode terdiri dari pembuatan model dari domain aplikasi, kemudian menambahkan detail implementasi pada saat desain dari suatu system

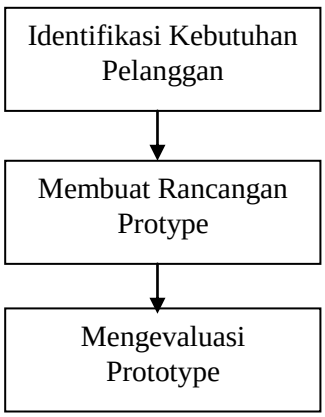
3.2.2.2 Metode Pengembangan Sistem

Metode pengembangan sistem menggunakan metode *prototype* yang merupakan metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai. Kadang-kadang pemakai atau *user* hanya memberikan beberapa kebutuhan umum *software* tanpa detail *input*, proses atau detail *output*.
Di lain waktu mungkin dimana tim pembangun (*developer*) tidak yakin terhadap efisiensi dari algoritma yang digunakan, tingkat adaptasi terhadap sistem operasi

atau rancangan form *user interface*. Ketika situasi seperti ini terjadi model *prototype* sangat membantu proses pembangunan *software*. Seperti pada semua metode, *prototype* dimulai dari pengumpulan kebutuhan.

Dengan perencanaan yang cepat akan dibentuk konstruksi dari *prototypenya*. *Prototype* ini dievaluasi oleh pelanggan dan digunakan untuk mengelola kembali kebutuhan dari perangkat lunak yang dikembangkan. Suatu proses iterasi terjadi, setelah *prototype* disesuaikan dengan kebutuhan pelanggan, sementara pihak pengembang makin mengerti keinginan pemakai.

Tahapan proses model *prototype*, bisa dijelaskan sebagai berikut:



Gambar 3.2 Tahapan Proses Model Prototype

Sumber : Roger S. pressman, 2002 dalam Hanif Al Fatta (2007:37)

1. Indentifikasi kebutuhan : *developer* dan pelanggan bertemu dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan berikutnya. Detail kebutuhan mungkin tidak dibicarakan disini, pada awal pengumpulan kebutuhan.

2. Pengembang : pengembangan prototype dilakukan dengan cepat dan membuat rancangan semua aspek *software* yang diketahui, dan rancangan ini menjadi dasar pembuatan *prototype*.
3. Mengevaluasi *prototype* : pelanggan mengevaluasi *prototype* yang dibuat dan digunakan untuk memperjelas kebutuhan software. Perulangan ketiga proses ini terus berlangsung hingga semua kebutuhan terpenuhi. *Prototype-prototype* dibuat untuk memuaskan kebutuhan pelanggan dan untuk memahami kebutuhan pelanggan lebih baik.

3.2.2.3 Alat Bantu Analisis dan Perancangan Berorientasi Objek

Analisis berorientasi objek (OOA) adalah metode analisis untuk memeriksa kebutuhan-kebutuhan dari perspektif kelas-kelas dan objek-objek yang ditemukan di *domain* masalah. Selama analisis dan tahap perancangan, pengembangan mempunyai dua tugas, yaitu :

1. Identifikasi kelas-kelas dan objek-objek pembentuk pada *domain* masalah, (disebut *abstraksi* masalah).
2. Menemukan struktur dimana himpunan-himpunan objek yang bekerja sama menyediakan perilaku-perilaku yang memenuhi kebuthuan-kebutuhan masalah (disebut mekanisme *implementasi*).

Dua bagian penting dari pengertian perancangan berorientasi objek adalah:

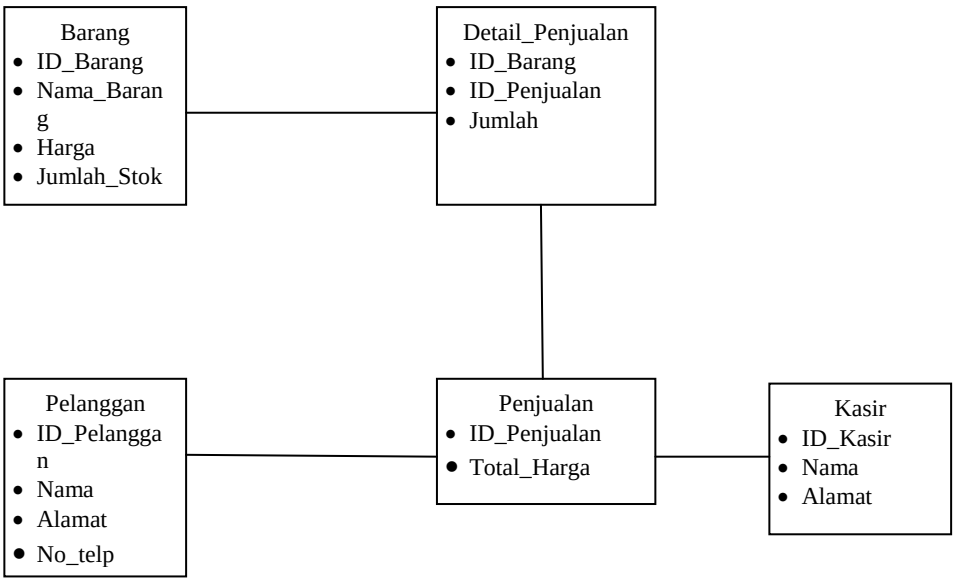
1. Menuntut menuju dekomposisi berorientasi objek
2. Menggunakan notasi-notasi berbeda untuk mengekspresikan model-model perancangan logik berbeda (struktur kelas dan objek) dan perancangan

fisik (arsitektur modul dan proses), serta juga model statis dan dinamis sistem yang sedang dirancang.

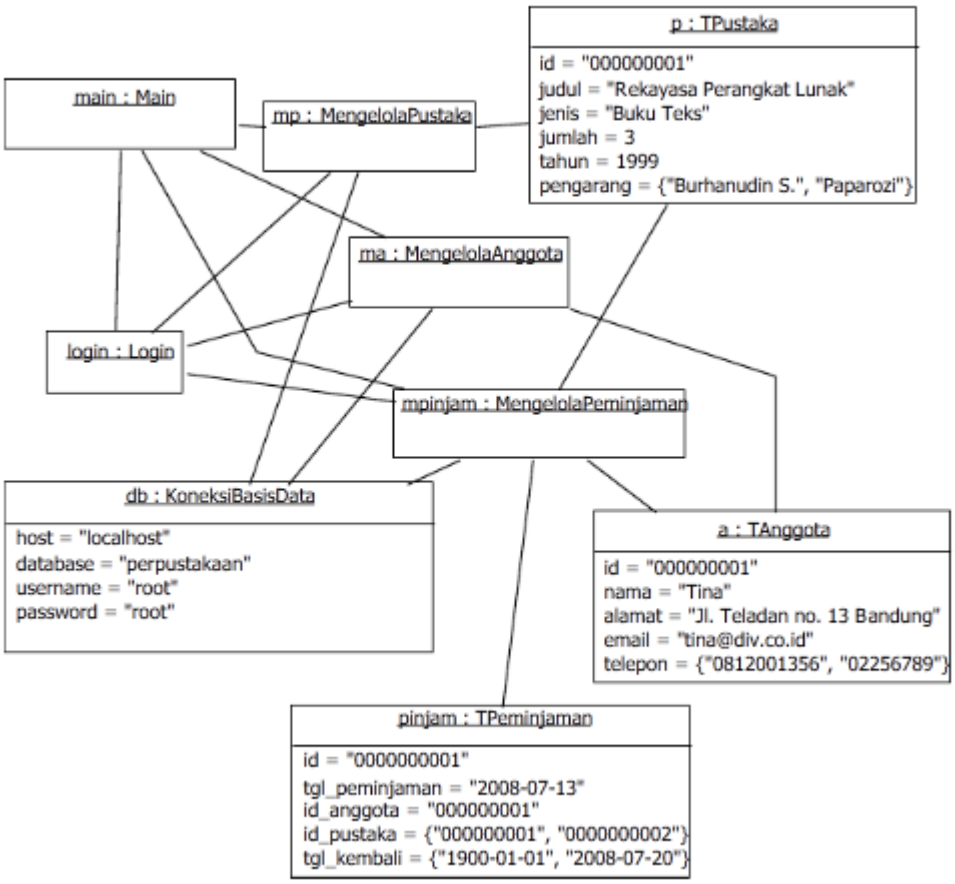
Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program dan digunakan luas dalam teknik piranti lunak skala besar. Lebih jauh lagi pendukung OOP *(Object Oriented Programming*) mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya dan pendekatan OOP lebih mudah dikembangkan dan dirawat. Dalam UML (*Unified Modelling Language*) yang dapat dikelompokan berdasarkan sifatnya yaitu statis atau dinamis jenis diagramnya adalah sebagai berikut :

1. Diagram Kelas (*Class Diagram*) bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan berorientsi objek.

Gambar 3.3. Gambaran Diagram Kelas (Class Diagram)

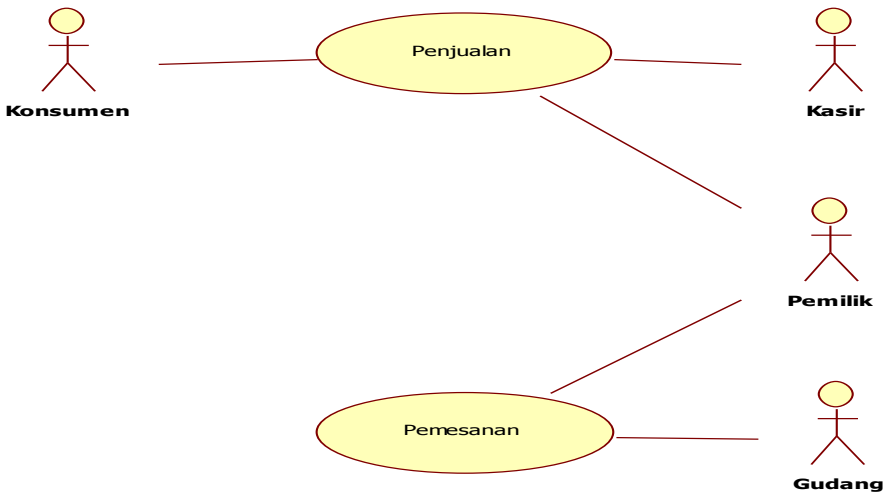


2. Diagram Objek (*Object Diagram*) bersifat statis. Diagram ini memperlihatkan objek-objek serta relasi-relasi antar objek. Diagram objek memperlihatkan instansiasi statis dari segala sesuatu yang dijumpai di dalam diagram kelas.



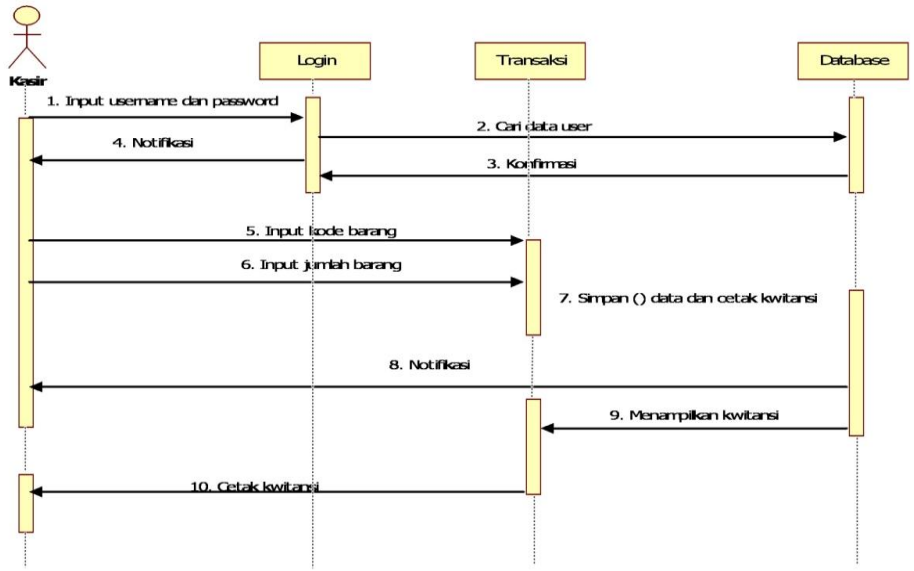
Gambar 3.3. Gambaran Diagram Objek (Object Diagram)

3. Diagram Use-Case (*Use-Case Diagram*) bersifat statis. Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna.



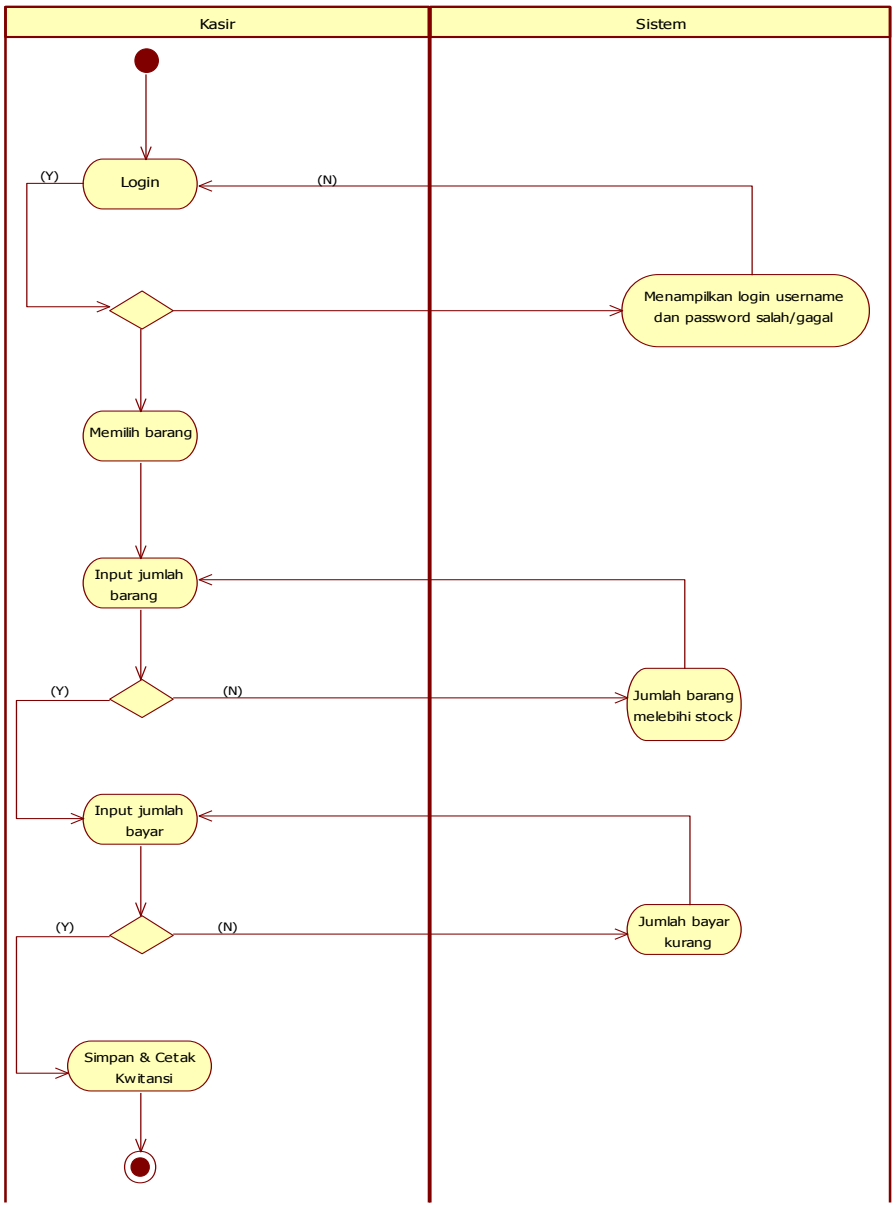
Gambar 3.4. Gambaran Diagram Use-Case (Use-Case Diagram)

4. Diagram Sekuen (*Sequence Diagram*) bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu.



Gambar 3.5. Gambaran Diagram Sekuen (Sequence Diagram)

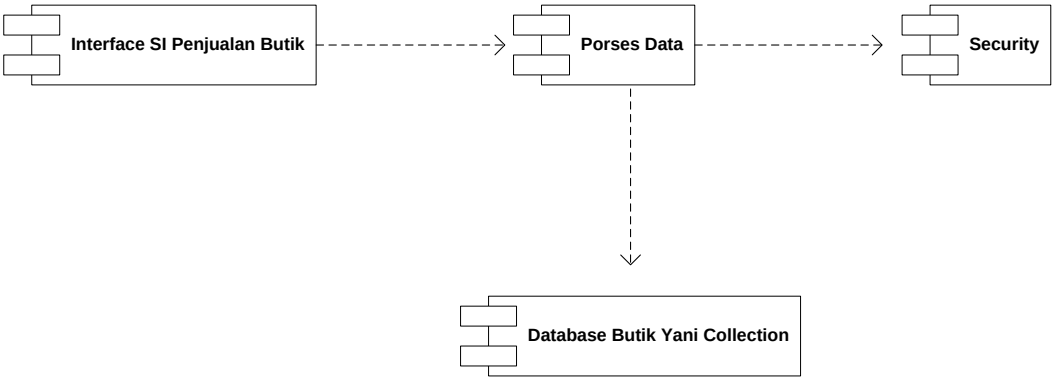
5. Diagram aktivitas, (*Activity Diagram*) bersifat dinamis. Diagram ini adalah diagram tipe khusus dari digram state yang memperlihatkan aliran dari suatu aktifitas ke aktifitas yan lainnya dalam suatu sistem. Diagram ini penting dalam pemodelan fungsi-fungsi dalam satu sistem dan memberi tekanan pada aliran kendali antar objek.



Gambar 3.6. Gambaran Aktivitas Diagram (Activty Diagram)

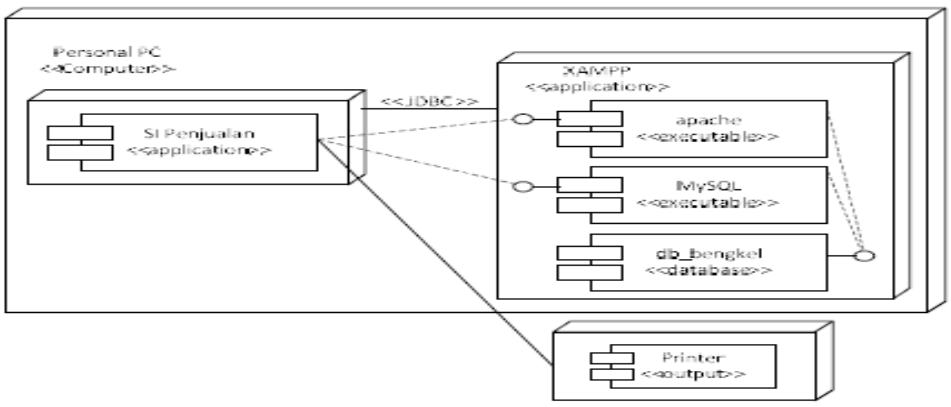
6. Diagram Komponen (*Componen Diagram*) bersifat statis. Diagram ini memperlihatkan organisasi serta ketergantungan system atau perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini

berhubungan dengan diagram kelas dimana komponen secara tifikal dipetakan kedalam satu atau lebih kelas-kelas.



Gambar 3.7. Gambaran Diagram Komponen (*Componen Diagram*)

7. Diagram Penyebaran (*Deployment Diagram*) bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. Deployment diagram berhubungan dengan diagram komponen dimana deployment diagram memuat satu atau lebih komponen-komponen.



Gambar 3.8. Gambaran Diagram Penyebaran (*Deployment Diagram*)

3.2.2.4 Pengujian Software

Dalam pembuatan suatu program baik berupa sistem informasi atau bukan, membutuhkan suatu teknik pengujian yang berguna dalam mengevaluasi program yang dibuat. Menurut Julian Supardi, M.T yang diterjemahkan oleh Rosa Ariani Sukamto : Black-Box Testing terfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. Black Box Testing bukanlah solusi alternatif dari White-Box Testing tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh White-Box Testing.

Konsep Kotak Hitam (*Black Box*) digunakan merepresentasikan sistem cara kerja di dalamnya tidak tersedia untuk diinspeksi. Di dalam Kotak Hitam (*Black Box),* item-item yang diuji dianggap ”gelap” karena logiknya tidak diketahui, yang diketahui hanya apa yang masuk dan apa yang keluar dari kotak hitam.

Pada pengujian Kotak Hitam (*Black Box*), kasus-kasus pengujian berdasarkan pada spesifikasi sistem. Rencana pengujian dapat dimulai sendiri mungkin di proses pengembangan perangkat lunak. Black-Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada
2. Kesalahan antarmuka (interface errors)
3. Kesalahan pada struktur data dan akses basis data
4. Kesalahan performansi (performance errors)
5. Kesalahan inisialisasi dan terminasi.

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

1. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan valid?
2. Input seperti apa yang dapat menjadi bahan kasus uji yang baik?
3. Apakah sistem sensitif pada input-input tertentu?
4. Bagaimana sekumpulan data dapat diisolasi?
5. Berapa banyak rata-rata data dan jumlah data yang dapat ditangani sistem?
6. Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem?