BAB 2
TINJAUAN PUSTAKA

2.1. Profil Tempat Kerja Praktek

Pada profil tempat kerja praktek yang diperoleh dari *Company profile* PD. BPR kota Bandung yang membahas mengenai sejarah perusahaan, logo perusahaan, badan hukum perusahaan, struktur organisasi dan *job description*, tempat dan kedudukan perusahaan, serta visi dan misi perusahaan.

2.1.1. Sejarah Perusahaan

PD. Bank Perkreditan Rakyat Kota Bandung didirikan pada tanggal 1 Agustus 1968 dengan Instruksi Walikotamadya Bandung, Nomor : 476/67/DPP dengan nama Bank Pasar Kotamadya Daerah Tingkat II Bandung yang statusnya merupakan seksi dari Dinas Pendapatan Daerah. Pada tanggal 22 Oktober 1996 PD. Bank pasar kotamadya Daerah Tingkat II Bandung berubah nama menjadi PD. BANK PERKREDITAN RAKYAT KOTAMADYA DAERAH TINGKAT II BANDUNG, sesuai Surat Keputusan Walikotamadya Bandung Nomor : 593/SK 642-BDG/1996 Beberapa PERDA yang terkait erat dengan operasional, Modal dasar dan Modal disetor PD BPR Kota bandung adalah:

1 PERDA Kotamadya Daerah Tingkat II Bandung No. 24 tahun 1994 tentang Perusahaan Daerah Bank Perkreditan Rakyat Kotamadya Daerah Tingkat II Bandung berisi perubahan nama bank menjadi Perusahaan Daerah Bank Perkreditan Rakyat. Perubahan Modal dasar menjadi Rp. 3.000.000.000,- dan Modal Disetor menjadi Rp.835.753.000,-.
2 PERDA Kotamadya Daerah Tingkat II Bandung No. 3 tahun 1999 tentang perubahan pertama PERDA No. 24 tahun 1994 dimana modal dasar bank dirubah menjadi Rp. 10.000.000.000,00 dan modal disetor sebesar Rp. 3.000.000.000,00.
3 PERDA Kota Bandung No. 21 tahun 2002 tentang BPR Kota Bandung dilakukan lagi perubahan Modal Dasar dari Rp. 10.000.000.000,00 menjadi Rp. 25.000.000.000,00
4 PERDA Kota Bandung No. 05 tahun 2005 tentang Perubahan kedua PERDA Kota Bandung No. 21 tahun 2002 dilakukan lagi perubahan Modal Dasar dari Rp. 10.000.000.000,00 menjadi Rp. 50.000.000.000,00

5 PERDA No.04 tahun 2009 tentang Perusahaan daerah Bank Perkreditan Rakyat Kota Bandung.[3]

2.1.2. Logo Perusahaan

Lambang kota Bandung ditetapkan dengan Peraturan Daerah Kota besar Bandung tahun 1953, tertanggal 8 Juni 1953, yang diijinkan dengan Keputusan Presiden tertanggal 28 april 1953 No. 104 dan diundangkan dalam Berita Propinsi Jawa Barat tertanggal 28 Agustus 1954 No. 4 lampiran No. 6. Lambang tersebut serupa dengan logo dari PD. BPR kota Bandung. Perisai tersebut terbagi dalam dua bagian oleh sebuah BALOK-LINTANG mendatar bertajuk empat buah, yang berwarna HITAM dengan pelisir berwarna PUTIH(PERAK) pada pinggir sebelah atasnya seperti pada gambar 2.1 yang akan dijabarkan sebagai berikut:

1. Bagian atas latar KUNING (EMAS) dengan lukisan sebuah GUNUNG berwarna HIJAU yang bertumpu pada blok-lintang.
2. Bagian bawah latar PUTIH(PERAK) dengan lukisan empat bidaang jalur mendatar berombak yang berwarna BIRU.

Di bawah perisai itu terlukis sehelai PITA berwarna KUNING (EMAS) yang melambai pada kedua ujungnya, Pada pita itu tertulis dengan huruf-huruf besar latin berwarna HITAM amsal dalam bahasa KAWI, yang berbunyi GEMAH RIPAH WIBAWA MUKTI. Sebagai tokoh lambang itu diambil bentuk perisai atau tameng, yang dikenal kebudayaan dan peradaban sebagai senjata dalam perjuangan untuk mencapai sesuatu tujuandengan melindungi diri. Perkakas perjuangan yang demikian itu dijadikan lambang yang mempunyai arti menahan segala mara bahaya dan kesukaran. Berikut ini adalah penjelasan makna setiap warna dan semboyan pada logo PD. BPR Kota Bandung:

1. KUNING (EMAS), berarti: kesejahteraan, keluhungan.
2. HITAM (SABEL), berarti: kokoh, tegak, kuat.
3. HIJAU (SINOPEL), berarti: kemakmuran sejuk
4. PUTIH (PERAK), berarti: kesucian
5. BIRU (AZUUR), berarti: kesetiaan
6. Gemah ripah wibawa mukti, berarti : tanah subur rakyat makmur.[3]



Gambar 2.1 Logo PD. BPR Kota Bandung

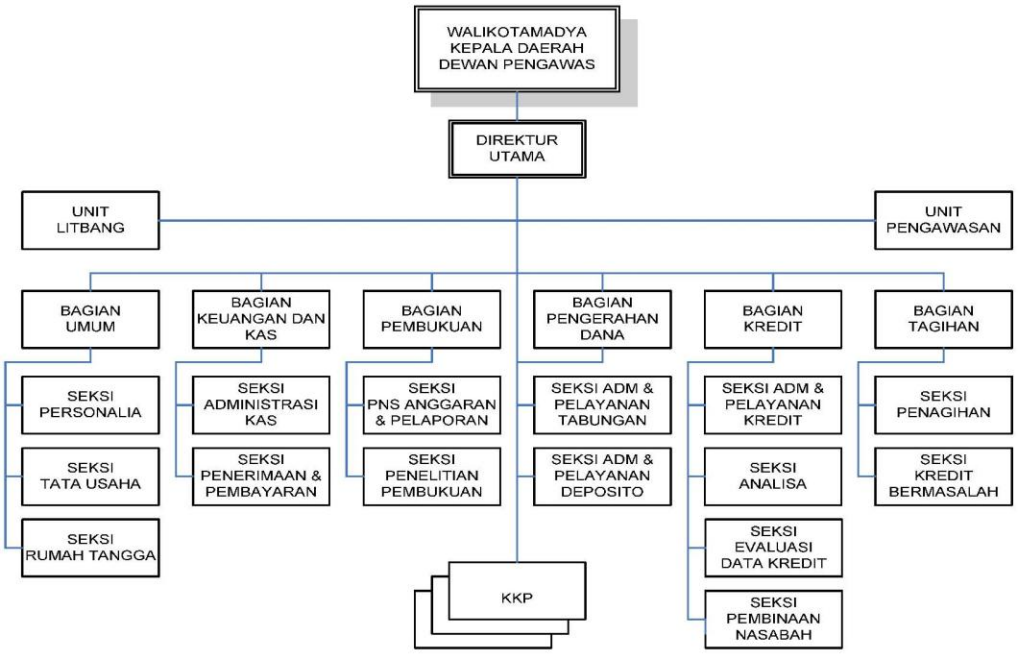
2.1.3. Badan Hukum Perusahaan

Dasar-dasar hukum yang mempertegas akan keberadaan dan eksistensi dari PD. BPRkota Bandung yang berdasarkan dari UU, PERDA Jawa Barat, instruksi menteri diantaranya adalah:

1. Undang-Undang Nomor 65 Tahun 1962 tentang Perusahaan Daerah.
2. Undang-Undang Nomor 05 Tahun 1974 tentang Pokok-Pokok Pemerintahan di Daerah.
3. Undang-Undang Nomor 07 Tahun 1992, diperbaharui dengan Undang-Undang Nomor10 Tahun 1998 tentang Perbankan.
4. Peraturan Menteri Dalam Negeri Republik Indonesia Nomor 04 Tahun 1993 tentang Perusahaan Daerah Bank Perkreditan Rakyat.
5. Peraturan Menteri Dalam Negeri Republik Indonesia Nomor 22 Tahun 2006 tentang Pengelolaan Bank Perkreditan Rakyat Milik Pemerintah Daerah.
6. Peraturan Bank Indonesia Nomor 6/22/PBI/2004 tanggal 09 Agustus 2004, diperbaharui dengan Peraturan Bank Indonesia Nomor 8/26/PBI/2006 tanggal 08 November 2006 tentang Bank Perkreditan Rakyat.
7. Keputusan Menteri Keuangan Republik Indonesia Nomor Kep.314/KM.17/ 1996 tentang Persetujuan Perubahan PD. Bank Perkreditan Rakyat Kodya Dati II Bandung.
8. Peraturan Daerah Kota Bandung Nomor 04 Tahun 2009 tentang Perusahaan Daerah Bank Perkreditan Rakyat Kota Bandung.[3]

2.1.4. Struktur Organisasi PD. BPR kota Bandung dan *Job Description*

Berdasarkan PERWAL Nomor 226 tahun 2007 tentang Organisasi dan Tata Kerja PD. BPR Kota Bandung, maka terbentuklah struktur organisasi PD. BPR kota Bandung seperti pada gambar 2.2 dan job description pada tabel 2.1[3].



Gambar 2.2 Struktur Organisasi PD. BPR kota Bandung

Tabel 2.1 *Job Description* PD. BPR kota Bandung

No	Jabatan	Deskripsi Perkerjaan
1.	Walikota Madya	Sebagai dewan pengawas dan penanggung jawab
2.	Direktur Utama	Sebagai pemegang kontrol perusahaan dan pelaksana tugas penanggung jawab harian
3.	Kepala Unit Litbang	Sebagai penanggung jawab bagian pelatihan dan pembangunan
4.	Kepala Unit Pengawasan	Mengawasi aktifitas dan mengontrol roda perusahaan
5.	Kepala Bagian	Mengepalai dan bertanggung jawab atas bagian dan subbagian masing-masing

2.1.5. Tempat dan Kedudukan Perusahaan

Lokasi PD. BPR kota Bandung beralamat di JL.Naripan No.29 Bandung.

2.1.6. Visi dan Misi Perusahaan

Sebuah perusahaan tentunya harus memiliki visi dan misi yang baik untuk meningkatkan citra, kinerja, tradisi agar menjadikan suatu perusahaan yang berbudaya.[3]

2.1.6.1. Visi Perusahaan

Visi dari PD. BPR kota Bandung adalah "Menjadi BPR yang unggul, handal, profesional, dan terpercaya". Jabaran visi:

1. PD. BPR kota Bandung didukung para pengelola yang mempunyai kompetensi dan integritas tinggi.
2. Operasional dikelola secara profesional dan manajemen berkualitas.
3. Peran intermediasi meningkat kepada sektor-sektor konsumtif.

PD. BPR kota Bandung didukung dengan kemampuan teknis mengenai sektor yang dibiayai dengan permodalan yang kuat, menghimpun sumber pendanaan dari masyarakat dan lembaga keuangan lain.[3]

2.1.6.2. Misi Perusahaan

Misi dari PD. BPR kota Bandung adalah "Tumbuh bersama untuk meningkatkan kesejahteraan dan perekonomian mayarakat serta memberikan kontribusi kepada *stakeholder* melalui jasa perbankan yang dikelola secara profesional". Jabaran misi:

1. Tumbuh Bersama

   Merupakan suatu ungkapan bahwa kota Bandung berkeinginan untuk selalu berkembang bersama-sama dengan masyarakat, pegawai maupun *stakeholder* lainnya..
2. Meningkatkan Kesejahteraan dan Perekonomian Mayarakat Jawa Barat

   Merupakan suatu keinginan untuk dapat memberikan manfaat kepada peningkatan kesejahtraan dan perekonomian nasabah maupun masyarakat pada umumnya di Jawa Barat.
3. Memberikan Kontribusi pada *Stakeholder*

   Merupakan suatu keinginan yang kuat dari PD. BPR kota Bandungkontribusi positif kepada PAD kota Bandung selaku pemilik dan juga kesejahteraan karyawan selaku *stakeholder*
4. Melalui Jasa Perbankan Yang Dikelola Secara Profesional

Merupakan suatu pembatasan aktivitas dari BPR Kota Bandung yang hanya akan melakukan aktivitas bisnis pada jasa perbankan yang dikelola secara profesional.[3]

2.2. Landasan Teori

Pada sub-sub bab landasan teori akan dijelaskan mulai dari definisi sistem, pendekatan sistem, karakteristik sistem, definisi inventori, definisi kontrol, dan definisi sistem inventori kontrol akan menjelaskan hal-hal yang berkaitan tentang sistem inventori kontrol.

2.2.1. Definisi Sistem

Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi, di mana suatu model matematika seringkali bisa dibuat. Pengertian dasar sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. [4]

2.2.1.1. Pendekatan Sistem

Pendekatan sistem dapat diartikan suatu cara berpikir yang menguraikan suatu sistem sebagai kerangka atau jaringan guna memperlihatkan faktor-faktor yang berbeda didalam dan diluar sistem sebagai suatu kumpulan yang terpadu.

Pendekatan sistem merupakan suatu persepsi tentang struktur yang mengkordinasikan bagian-bagian dan operasi-operasi dalam suatu organisasi atau sistem dengan cara yang efisien. [4]

2.2.1.2. Karakteristik Sistem

Sistem memiliki karakteristik sebagai berikut:

1. Memiliki komponen

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak perduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar

yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

2. Batas sistem (*boundary*)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*environment*)

   Lingkungan luar sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

4. Penghubung sistem (*interface*)

   Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.

5. Masukan sistem (*input*)

   Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam sistem komputer, program adalah *maintanance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran sistem (*Output*)

   Keluaran sistem merupakan hasil dari energi yang diolah oleh sistem.

7. Pengolah sistem (*Process*)

   Pengolahan sistem mmerupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran sistem

   Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. [4]

2.2.2. Definisi Inventori

Adapun menurut Sofjan Assauri (1993:169) inventori dapat didefinisikan sebagai berikut: "Inventori adalah suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha yang normal"[5].

Jadi inventori merupakan sejumlah barang yang disediakan untuk memenuhi permintaan dari pelanggan. Dalam perusahaan perdagangan pada dasarnya hanya ada satu golongan persediaan, yang mempunyai sifat perputaran yang sama yaitu yang disebut persediaan barang dagangan (*Merchandise Inventory)*. Persediaan ini merupakan persediaan barang yang selalu dalam perputaran, yang selalu dibeli dan dijual, yang tidak mengalami proses lebih lanjut didalam perusahaan tersebut yang mengakibatkan perubahan bentuk dari barang yang bersangkutan.

Persediaan pada dasarnya akan menimbulkan biaya-biaya. Biaya-biaya yang ditimbulkannya tersebut dapat berupa biaya tetap dan biaya *variable*. Menurut Bambang Rianto (1995) menyatakan bahwa untuk tujuan perencanaan besarnya persediaan kita hanya memperhatikan yang variabelnya saja dari biaya-biaya persediaan tersebut yang secara langsung akan terpengaruh oleh rencana tersebut[6].

Biaya Variabel dari persediaan tersebut dapat digolongkan menjadi:

1. Procurement (*Ordering Cost)*

   Ordering cost adalah biaya-biaya yang berubah-ubah sesuai dengan frekuensi pesanan, yang terdiri dari :

   a. Biaya selama proses pesanan, yaitu:
      i. Persiapan-persiapan yang diperlukan untuk pemesanan.
      ii. Penentuan besarnya kuantitas yang akan dipesan.
   b. Biaya pengiriman pesanan
   c. Biaya penerimaan barang yang dipesan, yaitu:
      i. Pembongkaran dan pemasukan ke gudang
      ii. Pemeriksaan material yang diterima
      iii. Mempersiapkan laporan penerimaan
      iv. Mencatat kedalam *Material Record Card*.[7]
   d. Biaya-biaya processing pembayaran, yaitu:
      i. Auditing dan perbandingan antara laporan penerimaan dengan pesanan yang asli.

ii. Persiapan pembuatan *cheque* untuk pembayaran.

iii. Pengiriman *cheque* dan kemudian auditnya.

2. *Carrying Cost*

Carrying cost adalah biaya yang berubah-ubah sesuai dengan besarnya persediaan. Penentuan besarnya carrying cost didasarkan pada persediaan rata-rata (*Average Inventory)*, dan biaya ini dinyatakan dalam persentase dari nilai dalam rupiah dari *average inventory*. Biaya-biaya yang termasuk kedalam *carrying cost* adalah :

a. Biaya penggunaan/sewa ruangan gudang
b. Biaya pemeliharaan material dan *allowances* untuk kemungkinan rusak
c. Biaya untuk menghitung atau menimbang barang yang dibeli
d. Biaya asuransi
e. Biaya modal
f. Biaya *absolescence*
g. Pajak dari persediaan yang ada dalam gudang.[7]

2.2.3. Definisi Kontrol

Kontrol adalah salah satu fungsi manajerial seperti perencanaan, pengorganisasian staf, dan mengarahkan. Kontrol merupakan fungsi penting karena dapat membantu untuk memeriksa kesalahan dan untuk mengambil tindakan korektif sehingga dapat meminimalkan penyimpangan dari standar dan tujuan organisasi tercapai dengan cara yang diinginkan.

Menurut konsep modern, kontrol adalah tindakan meramalkan bahwa konsep awal kontrol hanya digunakan ketika kesalahan terdeteksi. Pengendalian dalam pengelolaan berarti menetapkan standar, mengukur kinerja aktual dan mengambil tindakan korektif. Dengan demikian, kontrol terdiri dari tiga kegiatan utama.

Menurut Henri Fayol, kontrol merupakan melakukan pengendalian yang terdiri dari melihat sesuatu yang sedang dilakukan sesuai dengan rencana yang telah diadopsi, perintah yang telah diberikan, dan prinsip-prinsip yang telah ditetapkan. Tujuannya adalah untuk menunjukkan kesalahan agar dapat diperbaiki dan dicegah dari kesalahan berkali-kali.

Menurut Robert J. Mockler, kontrol dapat didefinisikan sebagai fungsi dari sistem yang menyesuaikan operasi yang diperlukan untuk mencapai rencana tersebut, atau untuk mempertahankan variasi dari tujuan sistem dalam batas-batas yang diijinkan. Fungsi subsistem kontrol dalam harmoni yang erat dengan sistem operasi. Sejauh mana kontrol berinteraksi

tergantung pada sifat dari sistem operasi dan tujuannya. Stabilitas menyangkut kemampuan sistem untuk mempertahankan pola *output* tanpa fluktuasi lebar. Kecepatan berkaitan dengan kecepatan respon dengan sistem yang dapat mengoreksi variasi dan kembali ke *output* yang diharapkan.[3]

2.2.4. Definisi Sistem Inventori Kontrol

Suatu sistem piranti lunak yang akan membantu proses inventarisasi dengan menerapkan tertib administrasi inventori yang ketat pencatatan dari barang masuk, penyimpanan, sampai dengan barang keluar. Dilengkapi dengan sistem pelaporan yang sistematis dan akurat.[5][6]

2.2.5. Metode Pengembangan Sistem

Metode yang digunakan pada sistem informasi ini adalah metode sekuensial linear atau sering juga disebut dengan metode *waterfall*. Dengan pengembangan sistemnya SDLC (*System Develompent Life Cycle*).[2]

2.2.5.1. Metode Waterfall

Model sekuensial linier untuk *software engineering*, sering disebut juga dengan siklus kehidupan klasik atau model air terjun seperti pada gambar 2.3. Model ini mengusulkan sebuah pendekatan kepada perkembangan yang sistematik 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:

1. Rekayasa dan pemodelan sistem/informasi (*System Engineering*)

Karena sistem merupakan bagian dari sebuah sistem yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa subset dari kebutuhan ke software tersebut.

Pandangan sistem ini penting ketika software harus berhubungan dengan elemen-elemen yang lain seperti software, manusia, dan database. Rekayasa dan anasisis sistem menyangkut pengumpulan kebutuhan pada tingkat sistem dengan sejumlah kecil analisis serta disain tingkat puncak. Rekayasa informasi mancakup juga pengumpulan kebutuhan pada tingkat bisnis strategis dan tingkat area bisnis.

2. Analisis kebutuhan software (*Analyst*)

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khusunya pada software. Untuk memahami sifat program yang dibangun, analis harus memahami domain informasi, tingkah laku, unjuk kerja, dan interface yang diperlukan. Kebutuhan baik untuk sistem maupun software didokumentasikan dan dilihat lagi dengan pelanggan.

3. Desain (*Design*)

Desain *software* sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Proses desain menerjemahkan syarat/kebutuhan ke dalam sebuah representasi *software* yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode. Sebagaimana persyaratan, desain didokumentasikan dan menjadi bagian dari konfigurasi *software*.

4. Generasi kode (*Coding*)

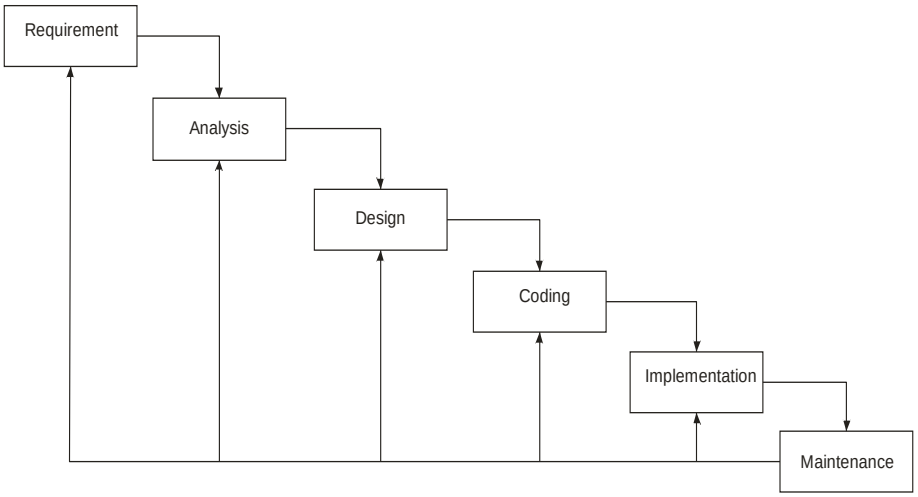
Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

5. Pengujian (*Testing*)

Sekali program dibuat, pengujian program dimulai. Proses pengujian berfokus pada logika internal software, 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 aktual yang sesuai dengan hasil yang dibutuhkan.

6. Pemeliharaan (*Maintenance*)

Software akan mengalami perubahan setelah disampaikan kepada pelanggan (perkecualian yang mungkin adalah software yangdilekatkan). Perubahan akan terjadi karena kesalahan–kesalahan ditentukan, karena software harus disesuaikan untuk mengakomodasi perubahan–perubahan di dalam lingkungan eksternalnya (contohnya perubahan yang dibutuhkan sebagai akibat dari perangkat peripheral atau sistem operasi yang baru), atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja. Pemeliharaan *software* mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.[2]



Gambar 2.2 Model *Waterfall*

2.2.6. Pemodelan dan Pemograman Prosedural

Pemodelan dalam suatu rekayasa perangkat lunak merupakan suatu hal yang dilakukan di tahapan awal. Di dalam suatu rekayasa dalam perangkat lunak sebenarnya masih memungkinkan tanpa melakukan suatu pemodelan. Namun, hal itu tidak dapat lagi dilakukan dalam suatu industri perangkat lunak. Pemodelan delam perangkat lunak merupakan suatu yang harus dikerjakan di bagian awal dari rekayasa, dan pemodelan ini akan mempengaruhi perkerjaan-pekerjaan dalam rekayasa perangkat lunak tersebut. Pemodelan terstruktur digunakan dalam analisis sistem, rancangan sistem, dan pengembangan perangkat lunak. Pada dasarnya merupakan pemodelan dengan pendekatan yang baku dan sistematik dengan menerapkan tahapan-tahapan yang sistematik dan bertujuan agar diperoleh hasil berupa suatu informasi yang bermanfaat dan memenuhi kebutuhan pemakai. Pemodelan ini merupakan pemodelan yang sudah lama dipakai. Istilah-istilah yang masih berkenaan dengan pemodelan prosedural diantaranya adalah:

1. Kamus Data (data *dictionary*): menyimpan deskripsi dari semua obyek data yang dikonsumsi atau diproduksi oleh perangkat lunak.
2. *Entity Relationship Diagram* (ERD): notasi yang digunakan untuk melakukan aktivitas pemodelan data.
3. *Data Flow Diagram* (DFD): untuk memberikan indikasi mengenai bagaimana data ditransformasi pada saat data bergerak melalui sistem, dan untuk menggambarkan fungsi-

fungsi serta subfungsi yang mentransformasi aliran data. Setiap fungsi disajikan dalam sebuah spesifikasi proses (PSPEC).

4. *State-transition* diagram (STD): menunjukkan bagaimana sistem bertingkah laku sebagai akibat dari kejadian eksternal dan digunakan sebagai dasar bagi pemodelan tingkah laku. Informasi tambahan mengenai aspek kontrol dari perangkat lunak diisikan dalam spesifikasi kontrol (CSPEC).

2.2.6.1. Pemodelan Data

Pemodelan data menjawab serangkaian pertanyaan spesifik yang relevan dengan berbagai aplikasi pemrosesan data. *Entity Relationship Diagram* (ERD) memungkinkan perekayasa perangkat lunak mengidentifikasi obyek data dan hubungan dengan notasi grafis. Pada konteks terstruktur, ERD menetapkan semua data yang dimasukan, disimpan, ditransformasi, dan diproduksi pada suatu aplikasi.

1. Obyek data, Atribut, dan Hubungan
   a. Obyek data adalah representasi dari hampir informasi gabungan yang harus dipahami oleh perangkat lunak. Obyek data dapat berupa entitas ekternal, benda, peristiwa, peran, unit oraganisasional, tempat, atau struktur. Obyek data digambarkan dengan empat persegi panjang.
   b. Atribut menentukan properti suatu obyek data dan mengambil salah satu dari tiga karakteristik yang berbeda. Atribut dapat digunakan untuk : menamai sebuah contoh dari obyek data, menggambarkan contoh, membuat referensi ke contoh lain pada tabel yang lain. Atribut adalah komponen untuk mendeskripsikan entitas yang digambar membentuk *elips*.
   c. Hubungan obyek data disambungkan satu dengan yang lainnya dengan berbagai macam cara. Hubungan obyek data dapat dibaca dari dua arah. Hubungan (relasi) adalah hubungan antara beberapa entitas yang digambarkan dengan belah ketupat. Relasi antar tabel dapat dikategorikan menjadi tiga macam yaitu: (1) *one to one relationship* (2) *one to many relationship* (3) *many to many relationship*.
   d. Simbol yang menyatukan atribut-atribut pada entitas tertentu serta menyatukan entitas-entitas dalam suatu relasi tertentu adalah garis.

2. *Entity Relationship Diagram*

ERD pada mulanya diusulkan oleh Peter Chen untuk desain sistem database relasional dan telah dikembangkan oleh yang lainnya. Serangkaian komponen utama diidentifikasikan oleh ERD: obyek data, atribut, hubungan, dan berbagai tipe indikator. Tujuan utama dari ERD adalah untuk mewakili obyek data dan hubungan mereka. Pemodelan data dan ERD memberi notasi yang singkat untuk mengamati data di dalam konteks aplikasi pemrosesan data kepada analisis yang digunakan untuk menciptakan suatu analisis, perancangan database dan untuk metode analisis persyaratan lainnya.

2.2.6.2. Pemodelan Fungsional dan Aliran Informasi

Informasi ditransformasikan pada saat mengalir melalui sebuah sistem berbasis komputer. Sistem tersebut menerima input dalam berbagai cara; mengaplikasikan perangkat keras dan elemen manusia untuk mentransformasikan input menjadi *output*; dan memproduksi *output* dalam berbagai bentuk.

2.2.6.2.1. Diagram Konteks

Diagram konteks menggambarkan hubungan antara sistem dengan entitas luarnya. Diagram konteks berfungsi sebagai transformasi suatu proses yang melakukan.

2.2.6.2.2. Data Flow Diagram

Andri Kristanto (2003:55), menjelaskan: "Data flow diagram adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data tersimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data tersimpan dan proses yang dikenakan pada data tersebut".

Adapun menurut Tata Sutabri (2003:163), "Data flow diagram adalah suatu network yangn menggambarkan suatu sistem automat/komputerisasi, manualisasi atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan.

Langkah-langkah dalam pembuatan DFD adalah sebagai berikut:

1. Diagram Konteks

Diagram yang menggambarkan sistem secara umum dari keseluruhan sistem yang ada.

2. Diagram level 1

Diagram yang menggambarkan arus data dalam proses secara lebih detail.

2.2.7. Basis Data

Database (basis data) merupakan kumpulan data yang saling berhubungan. Hubungan antar data dapat ditunjukan dengan adanya *field* atau kolom kunci dari tiap *file* atau tabel yang ada. Dalam satu *file* atau table terdapat *record-record* yang sejenis, sama besar, sama bentuk, yang merupakan satu kumpulan entitas yang seragam. Satu *record* (umumnya digambarkan sebagai baris data) terdiri dari *field* yang saling berhubungan menunjukan bahwa *field* tersebut dalam satu pengertian yang lengkap dan disimpan dalam satu *record*.

Sistem basis data adalah suatu sistem informasi yang mengintegrasikan dari kumpulan data yang saling berhubungan satu dengan yang lainnya dan membuat tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

Adapun struktur basis data adalah:

1. *Field*

   Suatu *field* mengambarkan suatu atribut dari *record* yang menunjukan suatu item dari data, seperti misalnya nama, alamat, dan lain sebagainya. Kumpulan dari *field* membentuk suatu *record*.

2. *Record*

   Kumpulan dari *field* membentuk suatu *record*. *Record* menggambarkan suatu unit data individu yang tertentu. Kumpulan dari *record* membentuk suatu *file*.

3. *File*

   File terdiri dari *recrd-record* yang menggambarkan satu kesatuan data yang sejenis.

4. *Database*

   Kumpulan dari *file* membentuk suatu database. Penyusunan suatu database digunakan untuk mengantasi suatu masalah penyusunan data, yaitu:

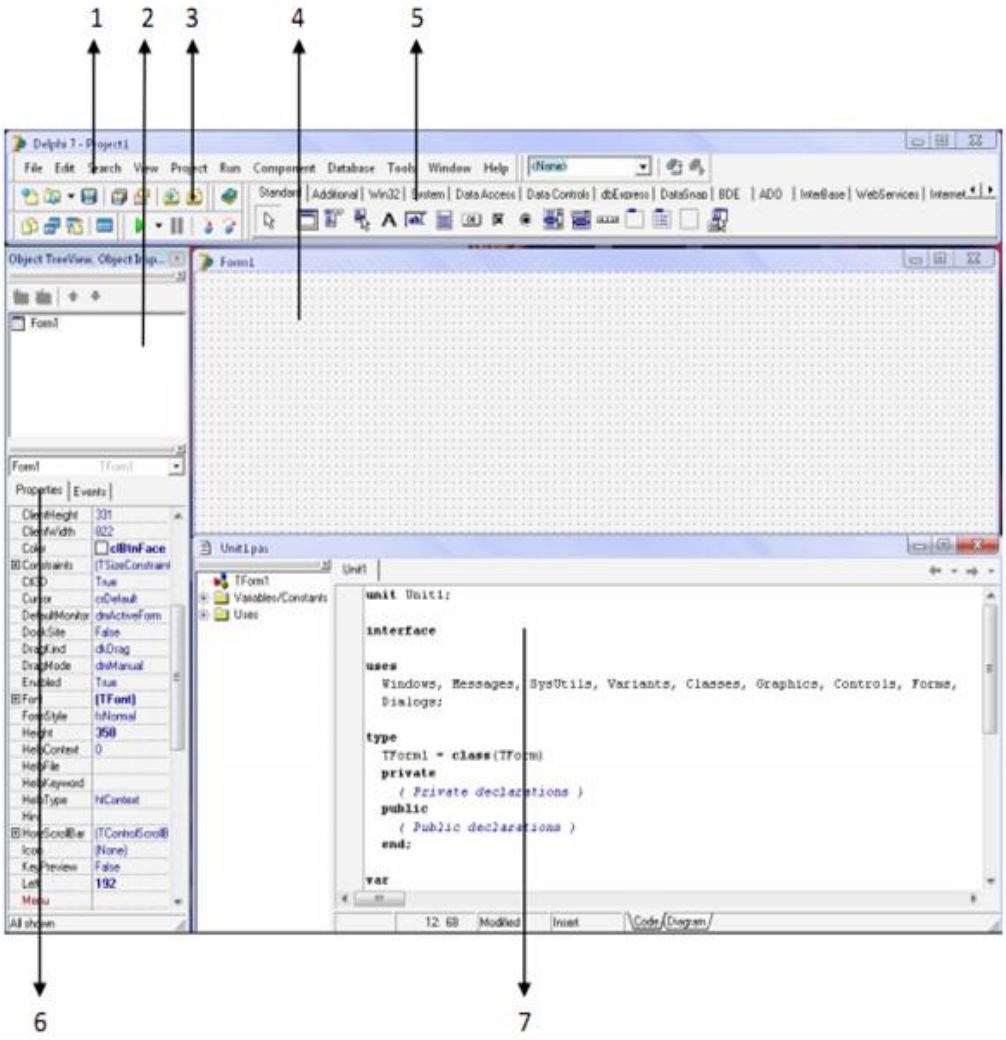
   a. Kesulitan pengaksesan
   b. Banyak pemakai (*multiple user*)
   c. Masalah keamanan (*security*)
   d. Masalah kesatuan (*Integration*)
   e. Masalah kebebasan data (*independence*)

2.2.8. Perangkat Lunak Pendukung

Software pendukung yang digunakan dalam pembangunan sistem inventori kontrol ATK dan cetakan PD. BPR kota Bandung adalah sebagai berikut.

2.2.8.1. Pengenalan *Borland Delphi* 7

Delphi merupakan salah satu bahasa pemograman visual yang dikembangkan oleh *Borland*. Bahasa pemrograman *Delphi* dapat digunakan untuk berbagai keperluan baik perhitungan matematis, aplikasi perkantoran, aplikasi multimedia, pembuatan aplikasi pengolah, dan lain-lain. Tampilan awal dan bagian-bagian dari *Delphi* 7 dapat dilihat dari gambar 2.4



Gambar 2.3 Bagian-bagian Borland Delphi

1. *Menu bar*

   Semua perintah yang diperlukan selama merancang dan membangun program aplikasi. Terletak di bagian atas jendela utama Delphi.

2. *Object TreeView*

   Berupa hirarki dari komponen-komponen pallets yang telah digunakan dalam program Delphi beserta nama dari kompone tersebut.

3. *Speed bar*

   Berisi sekumpulan tombol yang digunakan untuk mengakses beberapa perintah dalam menu. Biasanya yang tersedia pada *SpeedBar* adalah perintah-perintah yang umum digunakan dalam proses perancangan program aplikasi.

4. *Form Design*

   Merupakan tampilan visual dari aplikasi Delphi. Form berbentuk jendela dan dapat dianggap sebagai kertas atau meja yang digunakan untuk meletakkan komponen-komponen pallets. Saat memulai Delphi, akan otomatis tersedia sebuah form. Pada form tersebut terdapat garis titik-titik yang disebut *Grid*, berguna untuk membantu pengaturan tata letak obyek yang dimasukkan dalam form. Setiap form mengandung unit. Unit dalam form inilah yang dipakai untuk mengatur dan mengendalikan form.

5. Komponen *Pallets*

   Berisi kumpulan tab, dimana setiap tab/halaman memuat berbagai tombol komponen yang digunakan sebagai interface program aplikasi.

6. *Object Inspector*

   Object Inspector terdiri dari dua tab, yaitu:

   a. *Property*

      Digunakan untuk menentukan setting suatu obyek. Satu obyek memiliki beberapa properti yang dapat diatur langsung dari lembar properti pada jendela obyek inspector maupun melalui kode program. Setting ini mempengaruhi cara kerja obyek yang bersangkutan saat aplikasi dijalankan.

   b. *Event*

      Merupakan peristiwa atau kejadian yang diterima oleh suatu obyek, misalnya klik, drag, tunjuk dan lain-lain. *Event* yang diterima obyekakan memicu Delphi menjalankan kode program yang ada di dalamnya.

7. *Code Window*

   Merupakan tempat dimana kita mengetikkan kode program yang akan dieksekusi saat aplikai berjalan.[8]

2.2.8.2. Definisi *Microsoft Visio* 2007

Microsoft Visio (atau sering disebut *Visio*) adalah sebuah program aplikasi komputer yang sering digunakan untuk membuat diagram, diagram alir (*flowchart*), *brainstorm*, dan skema jaringan yang dirilis oleh *Microsoft Corporation*. Aplikasi ini menggunakan grafik vektor untuk membuat diagram-diagramnya.

Visio aslinya bukanlah buatan *Microsoft Corporation*, melainkan buatan *Visio Corporation*, yang diakusisisi oleh Microsoft pada tahun 2000. Versi yang telah menggunakan nama *Microsoft Visio* adalah *Visio* 2002, *Visio* 2003, dan *Visio* 2007 yang merupakan versi terbaru. *Visio* 2007 *Standard* dan *Professional* menawarkan antarmuka pengguna yang sama, tapi seri *Professional* menawarkan lebih banyak pilihan *template* untuk pembuatan diagram yang lebih lanjut dan juga penataan letak (*layout*). Selain itu, edisi *Professional* juga memudahkan pengguna untuk mengoneksikan diagram-diagram buatan mereka terhadap beberapa sumber data dan juga menampilkan informasi secara visual dengan menggunakan grafik.

Berikut ini adalah versi-versi *Visio*:

1. Visio 5.0
2. Visio 2000
3. Microsoft Visio 2002 (dikenal juga dengan sebutan Visio XP)
4. Microsoft Office Visio 2003
5. Microsoft 2007

Berikut ini adalah format berkas yang dapat ditangani *visio*:

1. `*.VSD`
2. `*.VSS`
3. `*.VST`
4. `*.vdx`
5. `*.vsx`
6. `*.vtx`

2.2.8.3. Definisi *PowerDesigner* 6

PowerDesigner adalah alat pemodelan kolaboratif perusahaan yang diproduksi oleh *Sybase*. *PowerDesigner* berjalan di bawah *Microsoft Windows* sebagai aplikasi asli, dan berjalan di bawah *Eclipse* melalui sebuah *plugin*. *PowerDesigner* mendukung desain arsitektur perangkat lunak *model-driven*. *PowerDesigner* menggunakan format *file* *.pdm.

2.2.8.4. Definisi *Database Desktop*

Database desktop adalah suatu alat dalam pembangunan suatu *database* yang disediakan langsung oleh *Delphi* 5 sampai dengan versi yang ke-7 dan merupakan sebuah *utility* yang disediakan untuk membuat tabel baru, membuat index, mengubah struktur tabel yang telah dibuat. *Database desktop* ini sendiri dapat juga dipakai untuk memasukan data ke dalam tabel. Tentunya dengan beberapa limitasi, khususnya yang berhubungan dengan *field* jenis *Memo* dan *binarry large object* (BLOB).[8]