BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Perusahaan

2.1.1 Sejarah PT. PLN (Persero)

Eksistensi PT PLN Distribusi Jawa Barat dan Banten memiliki sejarah panjang sebelumnya. Kelistrikan di tanah sunda sudah ada semenjak Pemerintah Kolonial Belanda masih bercokol di tataran bumi parahyangan. Di tahun 1905, di Jawa Barat khususnya kota Bandung, berdiri perusahaan yang mengelola penyediaan tenaga listrik bagi kepentingan publik. Nama perusahaan itu Bandungsche Electriciteit Maatschaappij (BEM).

Dalam perjalanannya, BEM pada tanggal 1 Januari 1920 berubah menjadi Perusahaan Perseroan menjadi Gemeenschapplijk Electriciteit Bedrijf Voor Bandoeng (GEBEO) yang pendiriannya dikukuhkan melalui akte notaris Mr. Andriaan Hendrik Van Ophuisen dengan Nomor: 213 pada tanggal 31 Desember 1949.

Setelah kekuasaan penjajahan beralih ke tangan Pemerintah Jepang, di antara rentah waktu 1942 - 1945, pendistribusian tenaga listrik dilaksanakan oleh Djawa Denki Djigyo Sha Bandoeng Shi Sha dengan wilayah kerja di seluruh Pulau Jawa.

Setelah Indonesia merdeka, tahun 1957 menjadi awal penguasaan pengelolaan penyediaan tenaga listrik di seluruh tanah air yang ditangani langsung oleh Pemerintah Indonesia.27 Desember 1957, GEBEO diambil alih oleh Pemerintah Indonesia yang kemudian dikukuhkan lewat Peraturan Pemerintah No. 86 Tahun 1958 j.o. Peraturan Pemerintah No. 18 Tahun 1959.

Selanjutnya, di tahun 1961 melalui Peraturan Pemerintah No. 67 dibentuk Badan Pimpinan Umum Perusahaan Listrik Negara (BPU-PLN) sebagai wadah kesatuan pimpinan PLN. Sejalan dengan itu, PLN Bandung pun berubah menjadi PLN Exploitasi XI sebagai kesatuan BPU-PLN di Jawa Barat, di luar DKI Jaya dan Tangerang.

Pada tahun 1970-an dikeluarkan Peraturan Pemerintah No. 18 Tahun 1972 tentang Perusahaan Umum Listrik Negara yang menyebutkan status PLN menjadi Perusahaan Umum Listrik Negara. Kemudian, berdasarkan Pengumuman PLN Exploitasi XI No. 05/DIII/Sek/1975 tanggal 14 Juli 1975, PLN Exploitasi XI diubah namanya menjadi Perusahaan Umum Listrik Negara Distribusi Jawa Barat.

Memasuki era 1990-an, dengan adanya Peraturan Pemerintah Republik Indonesia No. 23 Tahun 1994 pada tanggal 16 Juni 1994, Perusahaan Umum Listrik Negara Distribusi Jawa Barat diubah lagi menjadi Perusahaan Perseroan (Persero) dengan nama PT PLN (Persero) Distribusi Jawa Barat sejak tanggal 30 Juli 1994.

Untuk memenuhi tuntutan perubahan dan perkembangan kelistrikan yang dari tahun ke tahun cenderung mengalami peningkatan, maka keluarlah Keputusan Direksi PT PLN (Persero) No. 28.K/010/DIR/2001 tanggal 20 Februari 2001 yang menjadi landasan hukum perubahan nama PT PLN (Persero) Distribusi Jawa Barat menjadi PT PLN (Persero) Unit Bisnis Distribusi Jawa Barat.

Pada akhirnya, dengan mengacu pada Keputusan Direksi PT PLN (Persero) No. 120.K/010/DIR/2002 tanggal 27 Agustus 2002, PT PLN (Persero) Unit Bisnis Distribusi Jawa Barat berubah lagi namanya menjadi PT PLN (Persero) Distribusi

Jawa Barat dan Banten, di mana wilayah kerjanya meliputi Propinsi Jawa Barat dan Propinsi Banten, hingga saat ini.

2.1.2 Visi dan Misi PT. PLN (Persero)

visi dan misi yang terdapat pada PT PLN (Persero) Distribusi Jawa Barat dan Banten.

2.1.2.1 Visi Perusahaan

Visi Perusahaan yaitu diakui sebagai perusahaan kelas dunia yang tumbuh berkembang, unggul dan terpercaya dengan bertumpu pada potensi insani.

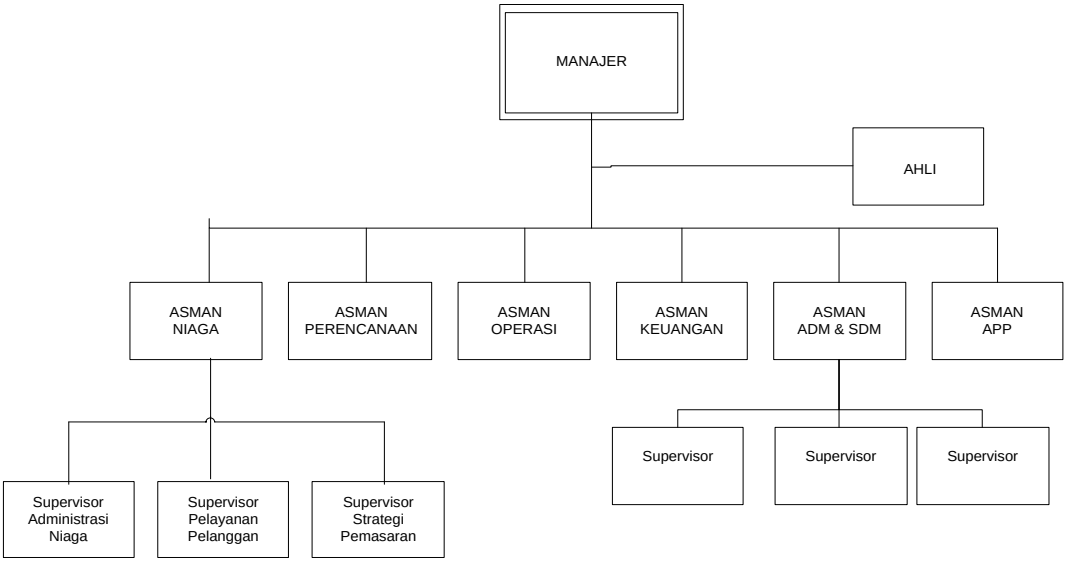
2.1.2.2 Misi Perusahaan

1. Menjalankan bisnis kelistrikan dan bidang usaha terkait yang berorientasi pada kepuasan pelanggan, karyawan dan pemegang saham.
2. Menjadikan tenaga listrik sebagai media untuk meningkatkan kualitas kehidupan masyarakat.
3. Mengupayakan agar tenaga listrik menjadi pendorong kegiatan ekonomi.
4. Menjalankan kegiatan usaha yang berwawasan lingkungan.

2.1.3 Struktur Organisasi

Struktur organisasi dapat dikatakan sebagai pola hubungan yang mapan diantara komponen-komponen atau bagian-bagian dari organisasi. Struktur organisasi adalah suatu kerangka yang mewujudkan suatu pola tetap dari hubungan antara kedudukan dan peranan dari suatu lingkungan kerjasama. Dan pada umumnya struktur organisasi berupa gambaran grafis yang menunjukkan hubungan antara unit-unit dalam organisasi serta garis-garis wewenang yang ada. Untuk mengetahui bagaimana susunan organisasi pada PT PLN (Persero) Distribusi Jawa Barat dan Banten APJ

Bandung beserta susunan jabatan dari masing-masing bidang dapat dilihat pada gambar 2.1.



Gamb Struktur Organisasi PT. PLN (Persero) Distribusi Jawa Barat dan Banten APJ Bandung

Berikut adalah *Job Description* dari PT. PLN (Persero) Distribusi Jawa Barat dan Banten APJ Bandung :

1. Manajer
    a. Mensinergikan seluruh unit garis depan dan seluruh fungsi di APJ dalam mengoptimalkan sumber daya dan kemitraan untuk memaksimalkan Kinerja Unit dan Citra perusahaan berdasarkan hokum dan ketentuan yang berlaku, termasuk surat kuasa dan kebijakan General Manager, termasuk pengembangan sistem informasi terintegrasi dan *online* dan pengembangan garis depan baru.
    b. Menjalin Komunikasi dan hubungan kerja internal dan eksternal yang efektif, mengembangkan dan memberdayakan seluruh potensi SDM

untuk meningkatkan budaya perusahaan (integritas, saling percaya, peduli dan pembelajar) dan *Good Corporate Government* (*responbility, accountability, fairness dan transparency*) disertai apresiasi dan pembinaan SDM.

c. Berkoordinasi dengan unit P3B terkait, APD (bila ada), Unit Distribusi lain (bila ada) dan APJ yang berbatasan.

d. Melengkapi pengaturan lebih lanjut (yang belum diatur oleh kantor distribusi), melaksanakan monitoring dan evaluasi/audit internal.

2. Ahli

a. Membuat rekomendasi solusi masalah dan konsep realistis untuk memaksimalkan kinerja APJ.

b. Melaksanakan kegiatan tertentu, bekerjasama dengan unit garis depan atau fungsi terkait, termasuk operasional lapangan, untuk memaksimalkan kinerja APJ dengan persetujuan manajer atau asisten manajer yang bersangkutan.

c. Bertanggung jawab kepada manajer APJ.

3. Asisten Manajer Niaga

Fungsi utama asisten manajer niaga adalah memfasilitasi unit garis depan dalam memaksimalkan kinerja melalui fungsi administrasi niaga, sistem pelayanan dan sistem pemasaran.

4. Asisten Manajer Perencanaan

Fungsi utama asisten manajer perencanaan adalah mengelola fungsi perencanaan terpadu, sistem teknologi informasi bekerjasama dengan ahli

dan fungsi terkait di APJ, untuk memfasilitasi unit garis depan dalam memaksimalkan kinerjanya.

5. Asisten Manajer Operasi

   Fungsi utama asisten manajer operasi adalah mengelola fungsi pengoperasian sistem jaringan, bekerjasama dengan ahli dan fungsi terkait di APJ, untuk memfasilitasi unit garis depan dalam memaksimalkan kinerjanya.

6. Asisten Manajer Keuangan

   Fungsi utama asisten manajer keuangan adalah mengelola fungsi keuangan, bekerjasama dengan ahli dan fungsi terkait di APJ, untuk memfasilitasi unit garis depan dalam memaksimalkan kinerjanya.

7. Asisten Perencanaan Sumber Daya Manusia dan Administrasi

   Fungsi utama asisten manajer SDM dan administrasi adalah mengelola fungsi SDM, fungsi administrasi, hokum dan komunikasi, logistic, bekerjasama dengan ahli dan fungsi di APJ, untuk memfasilitasi unit garis depan dalam memaksimalkan kinerjanya.

2.2 Landasan Teori

Pada bagian ini akan diuraikan teori-teori yang menunjang dalam pemecahan masalah yang dianggap relevan dengan pokok bahasan.

2.2.1 Pengertian 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.

Suatu sistem pada dasarnya merupakan suatu susunan yang teratur dari kegiatan yang berhubungan satu sama lain dan prosedur-prosedur yang berkaitan, yang melaksanakan dan mempermudah pelaksanaan kegiatan utama dari suatu organisasi. Menurut Hartono Jogiyanto, "Sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul, bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu"

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.

Suatu sistem mempunyai maksud tertentu. Ada yang menyebutkan maksud dari suatu sistem adalah untuk mencapai suatu tujuan (*goal*) dan ada yang menyebutkan untuk mencapai sasaran (*objective*). Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu diantaranya :

1. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-

komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem, atau bagian-bagian dari sistem.

2. Batas Sistem (*boundary*)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnnya 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 dari suatu sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*Interface*)

   Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem yang lainnnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Dengan penghubung, satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

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.

6. Keluaran Sistem (*Output*)

   Merupakan hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolah Sistem (*Process*)

   Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

8. Sasaran Sistem

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. [1]

2.2.1 Pengertian Informasi

Untuk menuju pada pengertian sistem informasi secara utuh, diperlukan pemahaman yang tepat tentang konsep data dan informasi. Keterkaitan data dan informasi sangatlah erat sebagaimana hubungan antara sebab dan akibat. Bahwa data merupakan bentuk dasar dari sebuah informasi, sedangkan informasi merupakan elemen yang dihasilkan dari suatu bentuk pengolahan data. Menurut Hartono

Jogiyanto, “Informasi adalah hasil dari pengolahan data menjadi bentuk yang lebih berguna bagi yang menerimanya, yang menggambarkan suatu kejadian nyata dan dapat digunakan sebagai alat bantu dalam pengambilan sebuah keputusan*”* . [1]

2.2.1.1 Kualitas Informasi

Kualitas dari suatu informasi sangat dipengaruhi atau ditentukan oleh 3 hal, yaitu :

1. Akurat (*accurate*)

   Sebuah informasi dapat dikatakan akurat jika informasi tersebut tidak bias atau menyesatkan, bebas dari kesalahan-kesalahan dan harus jelas mencerminkan maksudnya. Ketidakakuratan sebuah informasi dapat terjadi karena sumber informasi (data) mengalamai gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut.

2. Tepat waktu (*timeliness*)

   Sebuah informasi yang dihasilkan dari suatu proses pengolahan data, datangnya tidak boleh terlambat. Informasi yang terlambat tidak akan mempunyai nilai yang baik, sehingga kalau digunakan sebagai dasar dalam pengambilan keputusan dapat menimbulkan kesalahan dalam tindakan yang akan diambil.

3. Relevan (*relevance*)

   Informasi dikatakan berkualitas jika relevan bagi pemakainya. Informasi akan relevan jika memberikan manfaat bagi pemakainya.

2.2.2 Pengertian Sistem Informasi

Menurut Teguh Wahyono, "Sistem informasi diartikan sebagai suatu jaringan daripada beberapa elemen-elemen yang saling berhubungan serta membentuk satu kesatuan untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk beroperasi dengan cara yang sukses, serta member sinyal kepada manajemen dan yang lain terhadap kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan*"*. Istilah sistem informasi menyiratkan suatu pengumpulan data yang terorganisir beserta tata cara penggunaannya yang mencakup lebih jauh daripada sekedar penyajian. Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tata cara penggunaannya.

2.2.2.1 Komponen Sistem Informasi

Sistem informasi terdiri dari 6 komponen utama, yaitu :

1. Blok Masukan (*input block*)

   Blok ini terdiri dari data mentah informasi yang akan dijadikan sumber daya informasi.

2. Blok Model (*model block*)

   Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di dasar data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*output block*)

Produk keluaran yang dihasilkan sistem informasi berupa informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta user.

4. Blok Teknologi (*technology block*)

   Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, memperoleh dan mengirim output serta kontrol dari sistem secara keseluruhan komponen teknologi.

5. Blok Basis Data (*database block*)

   Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*control block*)

   Blok kendali adalah blok pengendali sistem informasi. Beberapa pengendali perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur, kesalahan-kesalahan dapat langsung cepat diatasi. [2]

2.2.3 Pengertian Kepegawaian

Kepegawaian berkaitan dengan penggunaan sumber daya manusia dalam suatu organisasi. Kepegawaian adalah fungsi yang mendukung terlaksananya fungsi HR yang lain. Secara umum fungsi ini bertanggung jawab terhadap *Employee Database*, *Payroll* dan pembayaran *benefit* lainnya, pinjaman karyawan, absensi, pencatatan cuti tahunan, dan lain-lain. Kepegawaian menangani evaluasi terhadap kinerja pegawai dengan aspek-aspek yang telah ditentukan oleh perusahaan yang terkait.

2.2.4 Pengertian Sistem Informasi Kepegawaian

Tiap perusahaan memiliki suatu sistem untuk mengumpulkan dan memelihara data yang menjelaskan sumber daya manusia, mengubah data tersebut jadi informasi dan melaporkan informasi itu kepada pemakai. Sistem ini dinamakan sistem informasi sumber daya manusia (Human Resouces Management System), atau HRIS.

Human Resources Information System (HRIS) ini dalam bahasa Indonesia adalah sistem informasi sumber daya manusia (SISDM) atau lebih dikenal dengan istilah sistem informasi manajemen kepegawaian (SIMPEG) sistem informasi manajemen kepegawaian (SIMPEG) yaitu berkenaan merancang format-format data kepegawaian dan mengatur sistem pengumpulan, pengolahan, penyimpanan dan pelaporan informasi kepegawaian yang terdiri dari: Data Pegawai, Data Jabatan, Data Mutasi, Data Rekruitmen dan lain-lain sehingga dapat dikelola informasi tentang kinerja pegawai, perencanaan kebutuhan pegawai, pembinaan dan pengembangan karirnya, kesejahteraan, serta pemberhentian atau pemensiunannya.[6]

2.2.5 Pengertian Basis Data (*database*)

Basis Data (*database*) adalah kumpulan file-file yang saling berhubungan, hubungan tersebut biasa ditunjukkan dengan kunci dari tiap file yang ada. Dalam satu file terdapat record-record yang sejenis, sama besar, sama bentuk, merupakan satu kumpulan entitas yang seragam. Satu entitas terdiri dari *field-field* yang saling berhubungan untuk menunjukkan bahwa field tersebut dalam satu pengertian yang lengkap dan direkam dalam satu record. Untuk menyebut isi dari *field* maka digunakan atribut atau merupakan judul dari satu kelompok entitas tertentu, misalnya entitas

nama barang menunjukkan entitas nama barang dari barang. Entitas dalah suatu objek yang nyata dan akan direkam.

Merancang *database* merupakan suatu hal yang sangat penting. Perancangan model konseptual perlu dilakukan disamping perancangan model fisik. Unsur-unsur konsep pembangun *database* adalah:

1. *Field* atau Atribut

   Field atau atribut adalah identitas yang mewakili satu jenis data.Misalnya *Field* nama pelanggan, alamat dan nomor tlp pada tabel data toko buku.

2. *Record*

   Record adalah kumpulan elemen yang saling terkait yang menginformasikan tentang suatu entity secara lengkap. Suatu *record* mewakili satu data atau Informasi tentang seseorang. Contoh: nomor pelanggan, nama pelanggan, alamat, kota, tanggal pinjam, tanggal kembali.

3. *File*

   File adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda data valuenya.

4. Tabel

   Tabel adalah sebuah file yang menampung data-data dalam kelompok tertentu.

2.2.6 Metode KPI (*Key Performance Indicators)*

KPI merupakan suatu performance metric yang secara nyata dan jelas terkait dengan sasaran strategis organisasi yang mampu mendorong organisasi menerjemahkan strateginya ke dalam terminologi yang bisa dikuantifikasi. Rancangan

KPI yang baik memberikan informasi yang dalam, jelas dan tajam mengenai kecenderungan suatu kinerja, sementara itu juga didukung oleh ketersediaan metric yang rinci. KPI yang tepat juga membantu apakah organisasi sudah melakukan hal yang benar dan mengetahui apa yang perlu perbaikan (improvement) atau penyesuaian.

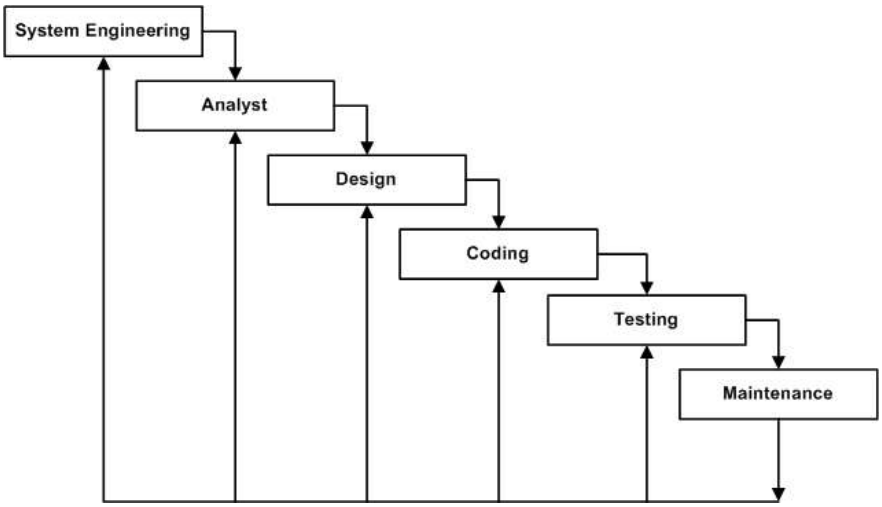
Dari namanya, *Key Performance Indicator* sudah menyebutkan, *performance indicator* atau penunjuk kinerja. Contohnya *performance* suatu proses diukur atau ditunjuk melalui suatu KPI. KPI bukan hanya mengukur suatu panjang, suatu waktu proses, suatu umur alat tetapi lebih tepat ukuran dari suatu *performance* atau kinerja. Lebih lanjut, KPI merupakan ukuran kunci (*key*) terhadap bisnis atau kesuksesan, bukan hanya ukuran seadanya / sambil lalu dari suatu bisnis proses. Dengan demikian, KPI sangat erat berhubungan dengan obyektif dari proses yang akan diukur. [3]

2.2.7 Metode Pengembangan Sistem

Metode yang digunakan pada sistem informasi ini adalah metode *waterfall*.dengan pengembangan sistemnya SDLC (sistem *develompent life cycle*), metode MRP (*material requirement planning*), dan metode garis-lurus (*straight-line method*).

2.2.7.1 Metode *Waterfall*

Model sekuensial linier untuk *software engineering*, sering disebut juga dengan siklus kehidupan klasik atau model air terjun. 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 yang dapat dilihat pada gambar 2.2 :



Gambar 2.1 Siklus Metode *Waterfall*

1. Rekayasa dan Pemodelan (Sistem Engginering)

   Rekayasa perangkat lunak merupakan tahapan yang pertama kali dilakukan untuk merumuskan sistem yang akan dibangun. Hal ini bertujuan untuk memahami sistem yang akan dibangun.

2. Analisis Perangkat Lunak (Software Analysis)

   Analisis dilakukan terhadap permasalahan yang dihadapi serta untuk menetapkan kebutuhan perangkat lunak dari aplikasi yang dibangun.

3. Perancangan perangkat Lunak (Software Design)

   Tahap desain merupakan tahap penerjemahan dari data yang telah dianalisis ke dalam bentuk yang mudah dimengerti oleh pengguna.

4. Implementasi perangkat lunak (Coding)

   Coding merupakan tahap penerjemahan data yang telah dirancang ke dalam bahasa pemrograman tertentu.

5. Pengujian perangkat lunak (Testing)

Tahap pengujian dilakukan terhadap perangkat lunak yang telah dibangun. Proses pengujian berfokus pada logika internal perangkat lunak serta memastikan apakah hasil yang diinginkan tercapai atau tidak.

6. Pemeliharaan (Maintenance)

   Maintenance merupakan penanganan dari suatu perangkat lunak yang telah selesai dibangun sehingga dapat dilakukan perubahan-perubahan atau penambahan sesuai dengan permintaan pengguna.

2.2.7.2 Metode *Object Oriented*

Pada metode pengembangan sistem ini akan menggunakan *object oriented* (OO) yang merupakan paradigma dalam rekayasa software yang didasarkan pada objek dan kelas. Object-oriented merupakan metodologi terbaik dalam rekayasa software dan mencakup bidang aplikasi yang sangat luas. Karena luasnya cakupan object-oriented makan terdapat beberapa hal yang membingungkan berkenaan dengan istilah dan konsep object-oriented. Istilah dan konsep yang berkenaan dengan object-oriented ini adalah :

1. *Object oriented Analysis* adalah metode analisis yang memeriksa requirement (syarat/keperluan yang harus dipenuhi suatu sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup permasalahan.
2. *Object oriented Design* adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

Object oriented analysis dan design merupakan pendekatan yang menekankan pada solusi logic berbasis objek. Beberapa konsep dasar dalam OOAD :

1. Objek

   Objek adalah “benda”, secara fisik atau konseptual dapat kita temui di sekeliling kita. Hardware, Software, dokumen, manusia dan bahkan konsep semuanya adalah konsep objek. Sebuah objek memiliki keadaaan sesaat *(state)* dan perilaku *(behaviour)*. State dari sebuah objek adalah kondisi objek tersebut atau himpunan dari keadaan yang menggambarkan objek tersebut.

2. Kelas

   Kelas adalah definisi umum untuk himpunan objek sejenis. Kelas menetapkan spesifikasi prilaku dan atribut objek-objek tersebut.

Object oriented merupakan metode yang paling baik dalam rekayasa perangkat lunak, diantaranya *procedure oriented*, *object-oriented*, *data struktur-oriented,* data *flow-oriented*, dan *constraint-oriented*. Sehingga dengan metode object-oriented ini dapat diaplikasikan dalam seluruh ruang lingkup rekayasa software.

Untuk memhami keunggulan OOAD maka kita harus memahami masalah yang dihadapi antara lain :

1. Software sulit untuk dimodifikasi bila memerlukan pengembangan.
2. Proses pembuatan software memerlukan waktu yang cukup lama sehingga kadang kala melebihi anggaran dalam pembuatannya.
3. Para programmer selalu membuat software dari dasar karena tidak adanya kode yang bias digunakan ulang (*reuse*).

Dari beberapa keunggulan diatas, dengan menggunakan object-oriented maka sangat menguntungkan bagi programmer karena programmer dapat mendesign program dalam bentuk objek-objek dan hubungan antara objek-objke tersebut untuk

kemudian dimodelkan dalam sistem nyata. Keuntungan yang lain adalah proses pembuatan software dapat dilakukan dengan lebih cepat karena software dibangun dalam objek-objek standard, sehingga dapat digunakan secara berulang-ulang.

Bahasa pemograman yang menggunakan *object oriented* diantaranya yang terkenal C++, Visual Basic dan Java.

2.2.8 Definisi UML (Unified Modeling Language) [4]

Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari tiga notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT

(*Object Modeling Technique*), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*).

2.2.8.1 Teknik-Teknik UML

Dalam tahap perancangan satu sistem diperlukan adanya teknik-teknik penyusunana sistem untuk menganalisa dan mempelajari objek yang ada unuk mengetahui apakah mereka dapat digunakan kembali atau diadaptasi untuk pemakaina baru atau menetukan satu objek baru atau yang dimodifikasi yang digabung dengan objek yang sudah ada ke dalam suatu aplikasi komputasi. Teknik-teknik tersebut adalah sebagai berikut:

1. **Use Case Diagram**

   Diagram ini menunjukan sekumpulan kasus fungsional dan actor (jenis kelas khusus) dan keterhubungannya diagram ini dapat sangat membantu bila kita sedang menyusun requitment sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada *sistem*.

2. **Sequence Diagram**

   Diagram ini menunjukan interaksi yang terjadi antar objek. Diagram ini merupakan pandangan dinamis terhadap sistem. Diagram ini menekankan pada basis keberurutan waktu dari pesan-pesan yang terjadi.

3. **Collaboration Diagram**

   Diagram ini merupakan diagram interaksi. Diagram ini menekankan pada organisasi stuktur dari objek-objek yang mengirim dan menerima pesan.

4. **Class Diagram**

Diagram ini adalah seebuah spesifikasi yang jika di instansiasi akan menghasilkan sebuah objek yang merupakan inti pengembangan dan desain yang berorientasi objek. Kelas mengambarkan keadaan suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut.

5. **Statechart Diagram**

   Statechart diagram menunjukkan transisi dan perubahan keadaan suatu objek pada sistem sebagai akibat dari stimulasi yang diterima. Dalam UML, state digambarkan berbentuk segi empat dengan sudut tumpul dan memiliki nama sesuai dengan kondisi saat itu.

6. **Activity Diagram**

   Actifity diagram menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses parallel yang mungkin erjadi pada beberapa eksekusi.

7. **Component Diagram**

   Component Diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan di antaranya.

8. **Deployment Diagram**

   Deployment diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, dimana komponen akan diletakkan (pada mesin, server atau piranti keras apa), bagaimana kemampuan

jaringan pada lokasi tersebut, spesifikasi server dan hal-hal lain yang bersifat fisikal.

2.2.9 Perangkat Lunak Pendukung

Pada bagian ini akan dijelaskan tentang beberapa perangkat lunak dan bahasa pemrograman yang digunakan untuk mendukung dalam pengembangan aplikasi yang dibuat. Bahasa pemrograman yang digunakan diantaranya adalah php dan html. Selain itu terdapat perangkat lunak yang digunakan yaitu Macromedia Dreamweaver dan SQL server.

2.2.9.1 Pengertian PHP [5]

PHP merupakan kependekan dari kata *Hypertext Prepocessor*. PHP tergolong sebagai perangkat lunak open source yang diataur dalam aturan *general purpose licences* (GPL).

Pemograman PHP sangat cocok dikembangkan dalam lingkungan web, karena PHP bisa dilekatkan pada *script* HTML atau sebaliknya. PHP dikhususkan untuk pengembangan web dinamis. Maksudnya, PHP mampu menghasilkan *website* yang secara terus menerus hasilnya bisa berubah-ubah sesuai dengan pola yang diberikan. Hal tersebut bergantung pada permintaan *client* browser yang digunakan. Pada umumnya, pembuatan web dinamis berhubungan erat dengan *database* sebagai sumber data yang akan ditampilkan.

PHP tergolong juga sebagai bahasa pemograman yang berbasis *server* (*server side cripting*). Ini berarti bahwa semua script PHP diletakkan diserver dan diterjemahkan oleh web *server* terlebih dahulu, kemudian hasil terjemahan itu dikirim

ke browser *client*. Tetu hal tersebut berbeda dengan *JavaScript* selalu tampak dihalaman web bersangkutan, jika dilakukan penyimpanan terhadap *file* web.

Secara teknologi, bahasa pemograman PHP memiliki kesamaan dengan bahasa ASP (*Active Server Page*), *Cold Fusion*, JSP (*Java Server Page*), ataupun Perl.

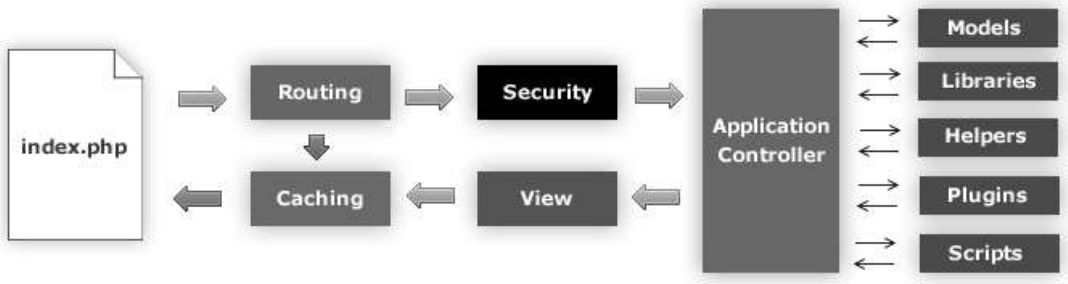
2.2.9.2 Pengertian *Codeigniter*

CodeIgniter adalah aplikasi open source yang berupa framework dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP. CodeIgniter memudahkan *developer* untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal.

2.2.9.3 Pengertian MVC (*Model, View,* dan, *Controller*)

MVC adalah kepanjangan dari *model, view,* dan *controller* yang merupakan merupakan suatu arsitektur dari aplikasi yang cakupannya diatas konsep OOP (*object oriented programming*) itu sendiri. Mengapa dapat dikatakan diatas cakupan OOP ? hal ini dikarenakan OOP mengatur dan menekankan bagaimana suatu kode ditulis maupun diterapkan sesuai dengan objek beserta atribut dan fungsinya. Dengan MVC maka penulisan struktur kode lebih rapih dan mempermudah pada saat dokumentasi, pengaturan tugas yang lebih terorganisir dimana model mengatur semua administrasi berhubungan dengan basis data baik *insert, update,delete* dan juga manipulasi data di basis data, view yang menyediakan antar muka antara sisi server dengan sisi client ataupum pengguna, dan terkahir *controller* yang mengendalikan dan mensinergikan antara *model* dan *view*. *Controller* bertugas untuk memanggil suatu objek atau class yang terdapat di dalam model maupun view.

Berikut adalah alur kerja dari MVC pada *codeigniter* yang terdapat pada gambar 2.2.



Gambar 2.2 Alur Kerja MVC

Yang dimana penjelasannya adalah sebagai berikut :

1. index.php berfungsi sebagai controller depan, mnginisialisasi basic resource yang dibutuhkah untuk menjalankan CI.
2. Router menganalisa HTTP request untuk menentukan apa yang harus dilakukan dengan HTTP request itu.
3. Jika file cache masih ada , maka akan dikirim langsung ke browser, tanpa melewati eksekusi normal sistem.
4. Keamanan, sebelum controller aplikasi di panggil, HTTP request dan data yang dikirim user, di filter untuk alasan keamanan.
5. Controller memanggil model, librari inti , plugin, helper, dan resource lainnya yang di butuhkan untuk memroses request tertentu.
6. View yang sudah diproses, dikirim ke browser sebagai hasil yang terlihat. Jika status caching ON, view akan disimpan di cache, jadi jika ada request yang sama, view itu bisa ditampilkan lagi.

2.2.9.4 Pengertian Macromedia Dreamweaver

Macromedia Dreamweaver merupakan salah satu software dari kelompok Macromedia yang banyak digunakan untuk mendesain situs Web.Adapun Macromedia Dreamweaver itu sendiri adalah sebuah HTML editor profesional untuk mendesain secara visual dan mengelolah situs atau halaman Web.

Dreamweaver 8 memiliki performa yang lebih baik dan memiliki tampilan yang memudahkan anda untuk membuat halaman web, baik dalam jendela desain maupun dalam jendela kode rumus. Dreamweaver 8 didukung dengan cara pemakaian yang praktis dan standar, dan juga didukung untuk pengembangan penggunaan CSS, XML, dan RSS, dan kemudahan-kemudahan lain yang diperlukan.

Dreamweaver merupakan software yang digunakan oleh Web desainer maupun Web programmer dalam mengembangkan Web.Hal ini disebabkan ruang kerja, fasilitas, dan kemampuan Dreamweaver yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun sebuah situs Web.

2.2.9.5 Pengertian Netbeans 6.8

NetBeans mengacu pada kedua platform kerangka untuk aplikasi desktop Java, dan lingkungan pengembangan terintegrasi (IDE) untuk pengembangan dengan Java, JavaScript, PHP, Python, Ruby, Groovy, C, C + +, Scala, Clojure dan masih banyak lagi.

NetBeans IDE ditulis dalam Java dan berjalan di mana-mana di mana JVM terinstal, termasuk Windows, Mac OS, Linux, dan Solaris. Sebuah JDK diperlukan untuk pengembangan fungsionalitas Jawa, tetapi tidak diperlukan untuk pembangunan di bahasa pemrograman lain. Platform NetBeans memungkinkan aplikasi untuk

dikembangkan dari serangkaian modular software komponen yang disebut modul. Aplikasi berbasis pada platform NetBeans (termasuk IDE NetBeans) dapat diperpanjang oleh pengembang pihak ketiga.

Versi terbaru saat ini adalah NetBeans IDE 6.8 yang mengembangkan fitur-fitur Java EE yang sudah ada (termasuk Java Persistence support, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans Enterprise Pack mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, web service dan pemodelan UML. NetBeans C/C++ Pack mendukung proyek C/C++.

Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali download, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga mengijinkan NetBeans untuk bisa dikembangkan. Fitur-fitur baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan. Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystem semuanya berbasis NetBeans IDE.

Paket-Paket Tambahan NetBeans IDE :

- NetBeans Enterprise
- Pack NetBeans Profiler
- NetBeans Enterprise Pack
- NetBeans Ruby Pack

• NetBeans JavaScript Editor

2.2.9.6 Pengertian MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia.Versi MySQL yang digunakan adalah MySQL Essential-5.0.24-win 32. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasional (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License) . Setiap pengguna dapat secara bebas menggunakan MySQL, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

MySQL memiliki beberapa kelebihan, diantaranya sebagai berikut :

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Multi-user. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

3. Performance tuning', MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
4. Ragam tipe data. MySQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
5. Perintah dan Fungsi. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (query).
6. Keamanan. MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenkripsi.
7. Skalabilitas dan Pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (records) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
8. Konektivitas. MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).
9. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.

10. Antar Muka. MySQL memiliki antar muka (interface) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).
11. Klien dan Peralatan. MySQL dilengkapi dengan berbagai peralatan (tool) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
12. Struktur tabel. MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostgreSQL ataupun Oracle.