BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Perusahaan

2.1.1. Sejarah Perusahaan

Lebih dari satu setengah abad yang lalu, tepatnya pada tahun 1852, di Cibodas Sindanglaya Cianjur, ditanam pohon kina (*Cinchona calisaya Wedd*) yang pertama di Indonesia, penanaman kina tersebut dilakukan oleh *Johannes Ellias Teysmann* yang ketika itu sebagai *Hortulanus* pada *'S Lands Planetentuin te Buitenzorg* (sekarang bernama Pusat Konservasi Tumbuhan Kebun Raya Bogor - LIPI).

Lokasi tersebut pada mulanya merupakan bagian dari Kebun Raya Bogor sebagai areal *aklimatisasi* (penyesuaiaan iklim) untuk jenis-jenis tanaman yang didatangkan dri luar negeri yang tidak dapat tumbuh baik di Bogor. Kemudian areal *aklimatisasi* tersebut dikembangkan menjadi kebun botani yang diberi nama *Bergtuin te Tjibodas* atau Kebun Pegunungan Cibodas.

Sesuai dengan surat keputusan ketua LIPI pada tanggal 17 Januari 1987 Nomor 25/KRP/D.5/87, kebun tersebut akhirnya diberi nama Cabang Balai Kebun Raya Cibodas.

2.1.2. Tugas Pokok Perusahaan

UPT Balai Konservasi Tumbuhan Kebun Raya Cibodas mempunyai tugas melakukan inventarisasi, ekplorasi, koleksi, penanaman dan pemeliharaan

tumbuhan pegunungan khususnya kawasan barat Indonesia yang memiliki nilai ilmu pengetahuan dan potensi ekonomi untuk dikoleksi dalam bentuk kebun botani, serta melakukan pendataan, pendokumentasiaan, pengembangan, pelayanan jasa dan informasi, pemasyarakatan ilmu pengetahuan dibidang konservasi, introduksi dan reintrodukasi tumbuhan.

2.1.3. Fungsi Perusahaan

Dalam melaksanakan tugas tersebut diatas, UPT Balai Konservasi Tumbuhan Kebun Raya Cibodas menyelenggarakan fungsi sebagai berikut :

1. Pelayanan, inventarisasi, eksplorasi, konservasi dan reintroduksi jenis tumbuhan dataran tinggi basah khususnya kawasan barat Indonesia yang memiliki nilai ilmu pengetahuan dan potensi ekonomi, pengembangan dan pendokumentasiaan biodata jenis tumbuhan koleksi yang berkaitan dengan konservasi Ex-Situ.
2. Pemberian pelayanan jasa ilmiah, pemasyarakatan ilmu pengetahuan dalam bidang konservasi tumbuhan dan introduksi tumbuhan.
3. Pelaksanaan urusan tata usaha dan rumah tangga.

2.1.4. Visi Perusahaan

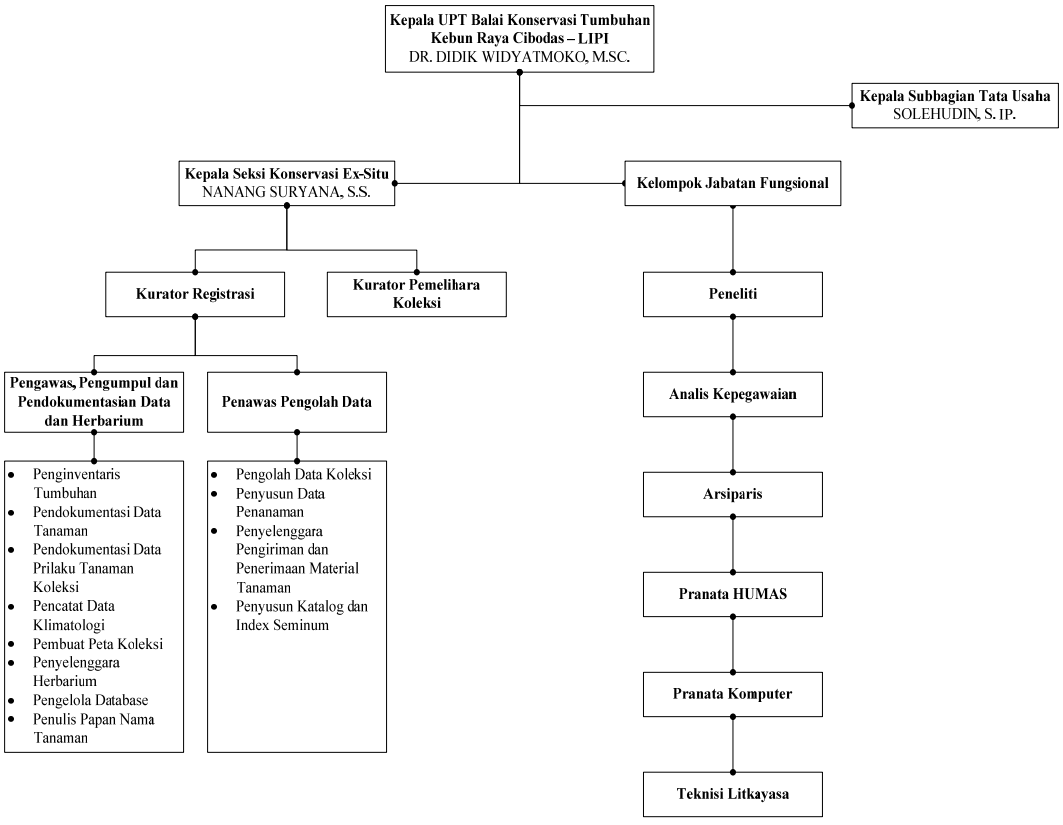
Menjadi salah satu Kebun Raya terbaik di dunia dalam bidang konservasi tumbuhan TROPIKA khususnya tumbuhan tropika dataran tinggi basah, penelitian, pelayanan pendidikan lingkungan dan pariwisata

2.1.5. Misi Perusahaan

Misi dari Kebun Raya Cibodas yaitu :

1. Melestarikan tumbuhan tropika khususnya tumbuhan tropika dataran tinggi basah.
2. Mengembangkan penelitian bidang konservasi dan pendayagunaan tumbuhan tropika khususnya tumbuhan tropika dataran tinggi basah.
3. Mengembangkan pendidikan lingkungan untuk meningkatkan pengetahuan dan apresiasi masyarakat terhadap tumbuhan dan lingkungan.
4. Meningkatkan kualitas pelayanan terhadap masyarakat.

2.1.6. Struktur Organisasi



Gambar 2.1 Struktur Organisasi Kebun Raya Cibodas

2.2. Landasan Teori

2.2.1. Sistem Informasi

2.2.2.1. Definisi Sistem Informasi

Ada beragam definisi sistem informasi yang didefinisikan oleh beberapa orang, tetapi dapat disimpulkan bahwa sistem informasi mencakup sejumlah komponen didalamnya (manusia, komputer, teknologi informasi, dan prosedur kerja), selain itu juga ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan.

Berikut adalah definisi sistem informasi menurut beberapa orang :

1. Gelinas, Oram dan Wiggins (1990) mengatakan bahwa sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada para pemakai
2. Alter (1992) mendefinisiskan sistem informasi sebagai kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.
3. Wilkinson (1992) mengungkapakan bahwa sistem informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan (*input*) menjadi keluaran (informasi), guna mencapai sasaran-sasaran perusahaan.

4. Bodnar dan Hopwood (1993) mengemukakan bahwa sistem informasi adalah kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna.
5. Turban, McLean, dan Wetherbe (1999) mendefinisikan sebuah sistem informasi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang sfesifik.
6. Hall (2001) menyampaiakan definisi sistem informasi adalah sebuah rangkaian prosedur formal di mana data dikelompokkan, diproses menjadi informasi, dan didistribusikan kepada pemakai.

2.2.2.2. Konsep Dasar Sistem dan Sistem Informasi

2.2.2.2.1. Sistem, Subsistem dan Supersistem

Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Berikut adalah beberapa elemen yang membentuk sebuah sistem :

1. Tujuan (*goal*)

Setiap sistem memiliki tujuan, entah hanya satu atau mungkin banyak. Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali. Tentu saja tujuan antara satu sistem dengan sistem yang lainya akan berbeda.

Begitu pula yang berlaku pada sistem informasi. Setiap sistem informasi memiliki suatu tujuan, tetapi dengan tujuan yang berbeda-beda.

Tujuan utama yang umum dari sistem informasi menurut Hall (2001) ada tiga macam, yaitu :

1) Untuk mendukung fungsikepengurusan manajemen,
2) Untuk mendukung pengambilan keputusan manajemen, dan
3) Untuk mendukung kegiatan operasi perusahaan.

Secara lebih sfesifik, tujuan system informasi bergantung pada kegiatan yang ditangani. Namun, kecenderungan penggunaan system informasi lebih ditujukan pada usaha menuju keunggulan kompetitif, yang artinya mampu bersaing dan mengungguli pesaing.

2. Masukan (*input*)

Masukan sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Pada sistem informasi, masukan dapat berupa data transaksi dan non-transaksi, serta intruksi.

3. Keluaran (*output*)

Keluaran merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

4. Proses (*process*)

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna, misalnya berupa informasi dan produk. Pada sistem informasi, proses dapat berupa suatu tindakan yang bermacam-macam. Meringkas data, melakukan perhitungan, dan mengurutkan data merupakan beberapa contoh proses.

5. Mekanisme pengendalian (*control mechanism*) dan umpan balik (*feed back*)

Mekanisme pengendalian diwujudkan dengan menggunakan umpan balik, yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan. Dalam bentuk sederhana, dilakukan perbandingan antara keluaran sistem dan keluaran yang dikehendaki (standar). Jika terdapat penyimpangan, maka akan dilakukan pengiriman masukan untuk melakukan penyesuaian terhadap proses supaya keluaran berikutnya mendekati standar. Bila penyebab penyimpangan terletak pada proses, maka proseslah yang diperbaiki. Pada sistem informasi, cara yang pertama dapat memberikan masukan pada setiap individu atau memberikan ringkasan kinerja terakhir untuk kegiatan manajemen. Adapun, hal yang terkahir sering terjadi pada sistem informasi karena program komputernyalah yang salah atau keluarannya dikehendaki untuk diubah.

Umpan balik seperti yang diutarakan di depan, yaitu menyesuaikan penyimpangan terhadap standar biasa disebut umpan balik negatif (*negative feedback*).

Selain dengan umpan balik negatif, pengendalian juga bisa memakai umpan balik positif (*positive feedback*) atau sering kali disebut umpan maju (*feedforward*). Pada sistem ini, pengendalian dimaksudkan untuk menambah kekuatan atau mendorong proses supaya memberikan hasil

yang lebih baik, tanpa harus menunggu terjadinya penyimpangan. umpan maju biasa digunakan untuk sistem yang mencegah terjadinya penyimpangan yang besar.

6. Batas (*boundary*)

Yang disebut batas sistem adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

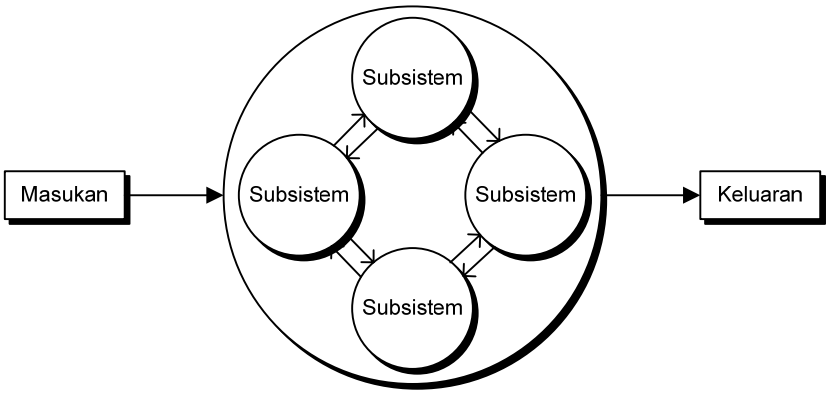
7. Lingkungan (*environment*)

Lingkungan adalah segala sesuatu yang berada di luar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri. Lingkungan yang merugikan tentu saja harus ditahan dan dikendalikan supaya tidak mengganggu kelangsungan operasi sistem, sedangkan yang menguntungkan tetap harus terus dijaga, karena akan memacu terhadap kelangsungan hidup sistem.

Sebuah sistem umumnya tersusun atas sejumlah sistem-sistem yang lebih kecil. Sistem-sistem yang berda dalam sebuah sistem itulah yang disebut subsistem. Jika sistem menjadi bagian dari sistem lain yang lebih besar, maka sistem yang lebih besar tersebut dikenal dengan sebutan supersistem.

2.2.2.2.2. Antarmuka Subsistem

Sebagaimana diketahui, sebuah sistem umumnya terdiri atas sejumlah subsistem. Masing-masing subsistem yang memiliki batas tersendiri ini saling berinteraksi untuk mencapai tujuan yang sama. Gambar 2.2 berikut ini merupakan gambaran dari antarmuka subsistem.



Gambar 2.2 Antarmuka subsistem

Dari gambar 2.2 di atas garis-garis yang menghubungakan antarsubsistem disebut antarmuka (*interface*) atau penghubung subsistem. Antamuka subsistem merupakan hal yang penting, sebab tanpa antarmuka ini sistem hanya berisi sekumpulan subsistem yang berdiri sendiri dan tidak saling berkaitan.

Secara prinsip, antarmuka subsistem berupa masukan dan keluaran. Dalam prakteknya, sebuah subsistem bisa saja hanya memberikan keluaran atau hanya menerima masukan. Pada prakteknya, sebuah antarmuka tidak sekedar menyatakan aliran data, melainkan juga melaksanakan suatu proses.

Menurut Martin (1999) dalam sistem informasi, pada dasarnya antarmuka berfungsi sebagai :

1. Penapisan

   Yakni membuang derau atau data yang tak berguna.

2. Pengkodean/pendekodean

   Yakni mengubah data dari suatu format ke dalam format yang lain.

3. Pendeteksian

   Yakni melakukan pemeriksaan dan pembetulan kesalahan-kesalahan terhadap standar atau kekonsistensian.

4. Penyanggatan

   Yakni memungkinkan dua buah sistem bekerja sama tanpa harus tersinkronisasi secara ketat. Caranya, antarmuka mengumpulkan data dari satu subsistem dan kemudian memperkenankan subsistem lain mengambil sata tersebut.

5. Pengamanan

   Yakni menolak permintaan yang berasal dari pihak yang tak berhak terhadap data dan menyediakan mekanisme proteksi yang lain.

6. Pengikhtisaran

   Yakni meringkas sejumlah masukan ke dalam bentuk agregat (ringkasan).

2.2.2.2.3. Klasifikasi Sistem

1. **Sistem Abstrak dan Sistem Fisik**

   Sistem abstrak (*abstract system*) adalah system yang berisi gagasan atau konsep. Misalnya, sistem teologi yang berisi gagasan tentang hubungan manusia dan Tuhan. Sedangkan sistem fisik (*physical system*) adalah sistem yang secara fisik dapat dilihat. Misalnya, sistem komputer.

2. **Sistem Deterministik dan Probabilistik**

   Sistem deterministik (*deterministic system*) adalah suatu sistem yang operasinya dapat diprediksi secara tepat. Misalnya, sistem komputer. Sedangkan sistem probabilistik (*probabilistic system*) adalah sistem yang tak dapat diramal dengan pasti karena mengandung unsur probabilitas. Misalnya, sistem arisan.

3. **Sistem Tertutup dan Terbuka**

   Sistem tertutup (*closed system*) adalah sistem yang tidak dapat bertukar materi, informasi, atau energi dengan lingkungan. Dengan kata lain, sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan. Misalnya, sistem penerimaan siswa baru (SPMB). Sistem terbuka (*open system*) adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Ciri-cirinya, sistem menerima masukan yang diketahui, yang bersifat acak, maupun gangguan. Selain itu, umumnya sistem melakukan adaptasi terhadap lingkungan. Misalnya, sistem perusahaan dagang.

4. Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah (*natural system*) adalah sistem yang terjadi karena alam (tidak dibuat oleh manusia). Misalnya, sistem tata surya. Sedangkan sistem buatan manusia (*human made system*) adalah sistem yang dibuat oleh manusia. Misalnya, sistem komputer.

5. Sistem Sederhana dan Sistem Kompleks

Berdasarkan tingkat kerumitannya, sistem dibedakan menjadi sistem sederhana (misalnya sepeda) dan sistem yang kompleks (misalnya otak manusia).

6. Kedudukan Sistem Informasi Sebagai Sistem

Sistem informasi tergolong sebagai :

1. Sistem buatan manusia,
2. Sistem terbuka,
3. Bersifat fisik, serta
4. Dapat tergolong sebagai sistem probabilistik atau deterministik (tergantung pada titik pandang untuk meninjaunya).

Sistem ini adalah sistem buatan manusia karena manusialah yang merancang dan mengoperasikannya. Tergolong sistem terbuka karena menerima masukan dan keluaran bagi lingkungan, serta beradaptasi dengan perubahan atau permintaan lingkungan. Termasuk sistem fisik karena secara fisik dapat dilihat. Sistem informasi dapat tergolong sebagai sistem kompleks atau sederhana tergantung pada implementasinya. Sistem informasi digolongkan sebagai sistem probabilistik karena kondisi masa

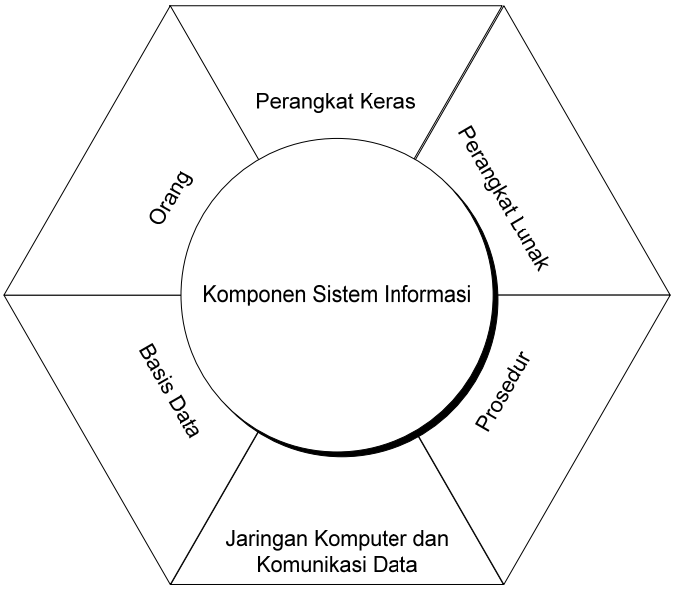
depan sistem ini tidak adapat diramalkan dengan pasti (Wilkinson, 1992). Namun, bagian tertentu seperti sistem pemrosesan transaksi dapat dianggap sebagai sistem yang deterministik.

2.2.2.3. Komponen Sistem Informasi

Dalam suatu sistem informasi terdapat komponen-komponen sebagai berikut :

1. Perangkat keras (*hardware*), mencakup peranti-peranti fisik seperti computer dan printer.
2. Perangkat lunak (*software*) atau program, yaitu sekumpulan intruksi yang memungkinkan perangkat keras untuk dapat memproses data.
3. Prosedur, yakni sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.
4. Orang, yaitu pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan, dan penggunaan keluaran sistem informasi.
5. Basis data (*database*), yaitu sekumpulan table, hubungan, dan lain-lain yang berkaitan dengan penyimpanan data.
6. Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sesumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

Gambar 2.3 berikut adalah gambar dari komponen sistem informasi :



Gambar 2.3 komponen sistem informasi

2.2.2.4. Jenis-Jenis Sistem Informasi

Ada berbagai cara untuk mengelompokkan sistem informasi. Klasifikasi umum yang dipakai antara lain didasarkan pada :

1. Level organisasi.
2. Area fungsional.
3. Dukungan yang diberikan
4. Arsitektur sistem informasi

Beberapa istilah sistem informasi lain juga sering dijumpai dalam literature, seperti sistem informasi strategis dan sistem informasi geografi. Sistem informasi geografis adalah sistem berbasis komputer yang digunakan untuk menyimpan dan memanipulasi informasi geografis (Aronoff, 1989).

2.2.2.4.1. Sistem Informasi Menurut Level Organisasi

Berdasarkan level organisasi, sistem informasi dapat dikelompokan menjadi :

1. Sistem informasi departemen, adalah sistem informasi yang hanya digunakan dalam sebuah departemen.
2. Sistem informasi perusahaan, adalah sistem informasi yang tidak terletak pada masing-masing departemen, melainkan berupa sebuah sistem terpadu yang dapat dipakai oleh sejumlah departemen secara bersama-sama.
3. Sistem informasi antarorganisasi, adalah sistem informasi yang menghubungkan dua organisasi atau lebih.

2.2.2.4.2. Sistem Informasi Fungsional

Sistem-sistem informasi ini terdiri dari berbagai sumber daya dalam organisasi yang tidak berdiri sendiri secara fisik, dalam sistem informasi perusahaan. Sistem-sistem informasi fungsional ini berkedudukan sebagai subsistem-subsistem. System informasi ini antara lain sebagai berikut :

1. Sistem informasi akuntansi, merupakan sistem informasi yang menyediakan informasi yang dipakai oleh fungsi akuntansi.
2. Sistem informasi keuangan, merupakan sistem informasi yang menyediakan informasi pada fungsi keuangan yang menyangkut keuangan perusahaan.

3. Sistem informasi manufaktur, merupakan sistem informasi yang bekerja sama dengan sistem informasi lain untuk mendukung manajemen perusahaan dalam menyelesaikan masalah yang berhubungan dengan produk atau jasa yang dihasilkan perusahaan.
4. Sistem informasi pemasaran, merupakan sistem informasi yang menyediakan informasi yang dipakai oleh fungsi pemasaran.
5. Sistem informasi SDM, merupakan sistem informasi yang dipakai oleh fungsi personalia.

2.2.2.4.3. Sistem Informasi Berdasarkan Dukungan yang Tersedia

Berdasarkan dukungan yang diberikan kepada pemakai, sistem informasi yang digunakan pada semua area fungsional dalam organisasi dapat diklasifikasikan sebagai berikut :

1. Sistem pemrosesan transaksi (*transaction processing system*).
2. Sistem informasi manajemen (*management infromation system*).
3. Sistem informasi perkantoran (*office automation system*).
4. Sistem informasi pendukung keputusan (*decision support system*).
5. Sistem informasi eksekutif (*executive information system*).
6. Sistem pendukung kelompok (*group support system*).
7. Sstem pendukung cerdas (*intelligent support system*).

2.2.2.4.4. Klasifikasi Menurut Aktivitas Manajemen

Sistem informasi terkadang diklasifikasikan berdasarkan aktivitas yang didukungnya pada level manajemen. Berdasarakan hal ini, terdapat pengelompokan sebagai berikut : sistem informasi pengetahuan, sistem informasi operasional, sistem informasi manajerial, dan sistem informasi strategis (Ebert dan Griffin, 2003).

Selain klasifikasi diatas terdapat pula klasifikasi menurut arsitektur sistem yaitu : sistem berbasis *mainframe*, sistem komputer pribadi (PC) tunggal, dan sistem tersebar atau sistem komputasi jaringan.

2.2.2. Model Proses Perangkat Lunak

2.2.2.1. Model Sekuensial Linier

Sekuensial linier mengusulkan sebuah pendekatan kepada perkembangan perangkat lunak 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.
2. Analisis Kebutuhan Perangkat Lunak.
3. Desain.
4. Generasi Kode.
5. Pengujian.
6. Pemeliharaan.

Model sekuensial linier adalah paradigma rekayasa perangkat lunak yang paling luas dipakai dan paling tua, masalah-masalah yang kadang-kadang terjadi ketika model sekuensial linier diaplikasikan adalah :

1. Jarang sekali proyek nyata mengikuti aliran sekuensial yang dianjurkan oleh model.
2. Kadang-kadang sulit bagi pelanggan untuk menyatakan semua kebutuhannya secara eksplisit.
3. Pelanggan harus bersikap sabar.
4. Pengembang sering melakukan penundaan yang tidak perlu.

Masing-masing dari masalah tersebut bersifat riil, tetapi paradigma siklus kehidupan klasik memiliki tempat yang terbatas namun penting di dalam kerja rekayasa perangkat lunak. Sekalipun memiliki kelemahan, secara signifikan dia lebih baik dari pada pendekatan yang sifatnya sembrono kepada pengembangan perangkat lunak.

2.2.2.2. Model Prototipe

Secara ideal prototipe berfungsi sebagai sebuah mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila prototipe yang sedang bekerja dibangun, pengembang harus mempergunakan fragmen-fragmen program yang ada atau mengaplikasikan alat-alat bantu yang memungkinkan program yang bekerja untuk dimunculkan secara cepat.

Prototipe bisa berfungsi sebagai sistem yang pertama. Para pemakai merasa enak dengan sistem aktual, sedangkan pengembang bisa membangunnya

dengan segera. Tetapi prototiping bisa juga menjadi masalah karena alasan-alasan sebagai berikut :

1. Pelanggan melihat apa yang tampak sebagai versi perangkat lunak yang bekerja tanpa melihat bahwa prototipe itu dijalin bersama-sama, tanpa melihat bahwa di dalam permintaan untuk membuatnya bekerja, kita belum mencantumkan kualitas perangkat lunak secara keseluruhan atau tanpa kemampuan pemeliharaan untuk jangka waktu yang panjang.
2. Pengembang sering membuat kompromi-kompromi implementasi untuk membuat prototipe bekerja dengan cepat.

Meskipun berbagai masalah bisa terjadi, prototipe bisa menjadi paradigma yang efektif bagi rekayasa perangkat lunak. Kuncinya adalah mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang keduanya harus setuju bahwa prototipe dibangun untuk berfungsi sebagai mekanisme pendefinisian kebutuhan.

2.2.2.3. Model RAD

Rapid Aplication Develoment (RAD) adalah sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi "kecepatan tinggi" dari model sekuensial linier di mana perkembangan cepat dicapai dengan menggunakan pendekatan kontruksi berbasis komponen.

Seperti semua proses model yang lain, pendekatan RAD memiliki kekurangan. Kekurangan dari model ini adalah sebagai berikut :

1. Bagi proyek yang besar tetapi berskala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim RAD yang baik.
2. RAD menuntut pengembang dan pelanggan memiliki komitmen di dalam aktifitas *rapid-fire* yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang sangat diperpendek. Jika komitmen tersebut tidak ada dari tiap konsituen, proyek RAD akan gagal.

Tidak semua aplikasi sesuai untuk RAD, RAD menekankan perkembangan komponen program yang bisa dipakai kembali.

2.2.2.4. Model Proses Perangkat Lunak Evolusioner

Model evolusioner adalah model literatif, model itu ditandai dengan tingkah laku yang memungkinkan perekayasa perangkat lunak mengembangkan versi perangkat lunak yang lengkap sedikit demi sedikit.

Model ini terdiri dari beberapa model, yaitu :

1. Model Pertambahan

   Model pertambahan atau incremental menggabungkan elemen-elemen model sekuensial linier dengan filosofi prototipe iteratif. Perkembangan pertambahan, khususnya berguna pada saat *staffing*, tidak bisa dilakukan dengan menggunakan implementasi lengkap oleh batas waktu bisnis yang sudah disepakati untuk proyek tersebut.

2. Model Spiral

   Model spiral adalah model proses perangkat lunak yang evolusioner yang merangkai sifat iteratif dari prototipe dengan cara control dan aspek

sistematis dari model sekuensial linier. Model itu berpotensi untuk pengembangan versi pertambahan perangkat lunak secara cepat. Di dalam model spiral, perangkat lunak dikembangkan di dalam suatu deretan pertambahan.

3. Model Rakitan Komponen

   Model ini menggabungkan beberapa karakteristik model spiral. Model ini bersifat evolusioner, sehingga membutuhkan pendekatan iteratif untuk menciptakan perangkat lunak.

4. Model Perkembangan Konkuren

   Model perkembangan konkuren sering digunakan sebagai paradigm bagi pengembangan aplikasi *client/server*. Sistem *client/server* dirancang dari serangkaian komponen fungsional. Bila diaplikasikan kepada *client/server* model proses konkuren akan mendefinisikan aktivitas di dalam dua dimensi yaitu, dimensi sistem dan dimensi komponen.

2.2.3. Basis Data

2.2.3.1. Definisi Basis Data

Basis data terdiri dari dua kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya.

Basis data sendiri dapat didefinisikan dalam sejumlahsudut pandang seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.

2.2.3.2. Database Management System (DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus. Perangkat lunak inilah yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. DBMS ini juga memberikan mekanisme pengamanan data, pemakaian data secara bersamaan, pemaksaan keakuratan/konsistensi data, dan lain sebagainya.

Perangkat lunak yang termasuk DBMS diantaranya yaitu dBase III+, dBase IV, FoxBase, Rbase, MS-Access, dan Borlan Paradox (untuk kelas sederhana) sedangkan Borland-Interbase, MS-SQLServer, CA-Open Ingres, Oracle, Informix dan Sybase (untuk kelas kompleks).

2.2.3.3. Bahasa Basis Data (*Database Language*)

DBMS merupakan perantara bagi pemakai dengan basis data dalam disk. Cara berinteraksi antara pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan pembuat DBMS. Bahasa itu dapat kita sebut Bahasa Basis Data yangterdiri atas sejumlah perintah (*statement*) yang diformulasikan dan dapat diberikan *user* dan dikenali/diproses oleh DBMS untuk melakukan suatu aksi/pekerjaan tertentu. Contoh-contoh bahasa basis data adalah SQL, dBase, QUEL dan sebagainya.

Sebuah bahasa basis data biasanya dapat dipilah ke dalam dua bentuk, yaitu :

1. *Data Definition Language* (DDL) dan
2. *Data Manipulation Language* (DML).

2.2.3.3.1. Data Definition Language (DDL)

Struktur/skema basis data yang menggambarkan/mewakili desain basis data secara keseluruhan dispesifikasikan dengan bahasa khusus yang disebut *Data Definition Language* (DDL). Dengan bahasa inilah kita dapat membuat tabel baru, membuat indeks, mengubah tabel, menentukan struktur penyimpanan tabel, dan sebagainya. Hasil dari kompilasi perintah DDL adalah kumpulan tabel yang disimpan dalam file khusus yang disebut Kamus Data (*Data Dictionary*).

Kamus data merupakan suatu metadata (superdata) yaitu data yang mendeskripsikan data yang sesungguhnya. Kamus data ini akan selalu

diakses dalam suatu operasi basis data sebelum suatu file data yang sesungguhnya diakses.

2.2.3.3.2. Data Manipulation Language (DML)

Merupakan bentuk bahasa basis data yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu basis data. Manipulasi data dapat beupa :

1. Penyisipan/Penambahan data baru ke suatu basis data.
2. Penghapusan data dari suatu basis data.
3. Pengubahan data di suatu basis data.

Data Manipulation Language (DML) merupakan bahasa yang bertujuan memudahkan pemakai untuk mengakses data sebagaimana direpresentasikan oleh model data. Ada dua jenis DML yaitu :

1. Prosedural, yang mensyaratkan agar pemakai menentukan data apa yang diinginkan serta bagaimana cara mendapatkannya.
2. Nonprosedural, yang membuat pemakai dapat menentukan data apa yang diinginkan tanpa menyebutkan bagaimana cara mendapatkannya.

2.2.4. Metode Perancangan Sistem

2.2.4.1. Diagram Aliran Data

Pada saat informasi mengalir melalui perangkat lunak, dia dimodifikasi oleh suatu deretan transformasi. Diagram aliran data/*data flow diagram* (DFD) adalah sebuah teknik grafis yang menggambarkan aliran informasi dan

transformasi yang diaplikasikan pada saat data bergerak dari input menjadi output. DFD tingkat o, yang disebut juga dengan model sistem fundamental atau diagram konteks, merepresentasikan seluruh elemen sistem sebagai sebuah bubble tunggal dengan data *input* (masuk) dan *output* (keluar) yang ditunjukan dengan anak panah yang masuk dan keluar secara berurutan.

2.2.4.2. Model Entity-Relationship (Model Keterhubungan-Entitas)

Pada Model Entity-Relationship, semesta data yang ada didunia nyata diterjemahkan dengan memanpfaatkan sejumlah perangkat konseptual menjadi sebuah diagram data, yang umum disebut sebagai Diagram Entity-Relationship (Diagram E-R). ada dua komponen utama pembentuk Model Entity-Relationship, yaitu Entitas (*Entity*) dan Relasi (*Relation*). Kedua komponen ini dideskripsikan lebih jauh melalui sejumlah Atribut/Properti.

Komponen Model Keterhubungan-Entitas lebih jelas adalah sebagai berikut :

1. Entitas (*Entity*) dan Himpunan Entitas (*Entity Sets*)

   Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain, sekelompok entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah Himpunan Entitas (*Entity Set*).

2. Atribut (*Atributes/Properties*)

   Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (*Properti*) dari entitas tersebut.

3. Relasi (*Relationship*) dan Himpunan Relasi (*Relationship Sets*)

   Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda, kumpulan dari semua relasi di antara entitas-entitas yang terdapat pada himpunan entitas tersebut membentuk himpunan relasi (*Relationship Sets*).

4. Kardinalitas/Derajat Relasi

   Kardinalitas relasi menunjukkan jumlah maksimum entitas yang sapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dapat berupa :

   a. Satu ke Satu (*One to One*)

      Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B.

   b. Satu ke Banyak (*One to Many*)

      Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, di mana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

c. Banyak ke Satu (*Many to One*)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, di mana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

d. Banyak ke Banyak (*Many to Many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, di mana setiap entitas pada himpunan entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

2.2.4.3. Diagram Entity-Relationship (Diagram E-R)

Model Entity-Relationship yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari dunia nyata yang kita tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan diagram entity-relationship (Diagram E-R). Notasi-notasi simbolik di dalam diagram E-R yang dapat kita gunakan adalah :

1. Persegi panjang, menyatakan himpunan entitas.
2. Lingkaran/elips, meyatakan atribut (Atribut yang berfungsi sebagai *key* digarisbawahi).

3. Belah ketupat, menyatakan himpuanan relasi.
4. Garis, sebagai penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.
5. Kardinalitas relasi dapat dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka (1 dan 1 untuk relasi satu-ke-satu, dan N untuk relasi satu-ke-banyak atau N dan N untuk relasi banyak-ke-banyak).

2.2.5. Perangkat Lunak Pendukung

Sistem informasi data tumbuhan ini dibangun menggunakan *Borland Delphi* sebagai antarmukanya dan *Microsoft SQL Server 2000* sebagai databasenya, untuk itu perlu dijelaskan sedikit tentang *Borland Delphi 7* dan *Microsoft SQL Server* ini.

2.2.5.1. Sekilas Tentang Borlad Delphi 7

Borland Delphi merupakan bahasa pemograman visual yang sudah terbukti keunggulannya dalam pengembangan aplikasi berbasis Windows. Borland Delphi merupakan pengembangan program Pascal yang merupakan bahasa pemograman yang sudah tidak asing lagi bagi para *programmer*. Dengan menggunakan konsep GUI (*Graphical User Interface*), sangat mudah bagi para *programmer* untuk merancang tampilan atau *user interface* program aplikasi yang akan dibuat. Pada dasarnya konsep pemograman berbasis Windows yang bersifat visual bertujuan untuk memudahkan pembuatan program. Konsep inilah yang

diimplementasikan oleh *software-software* pembuat program, antara lain Borland Delphi.

Salah satu keistimewaan Delphi adalah dukungannya untuk aplikasi berbasis *database*. Delphi 7 menambahkan fitur-fitur baru dan memperbaiki fitur-fitur yan gtelah dimiliki oleh Delphi 6, Perubahan-perubahan yang ada dapat ditemui di kategori IDE, Web, Com, *Database*, Component library, Runtuime library, Compiler, Rave Report, ModelMaker, dan Documentation.

Pada Borland Delphi kita dapat mengembangkan aplikasi-apliksi diantaranya :

1. Pengembangan aplikasi umum
2. Pengembangan aplikasi basisdata
3. Pengembangan aplikasi web
4. Pemograman untuk bermacam mesin / *platform*.

2.2.5.2. Microsoft SQL Server

Pada tahun 1988 Microsoft mengeluarkan SQL Server yang dikembangkan bersama Sybase. Awalnya SQL Server dirancang untuk dapat berjalan pada *platform* OS/2.

Selama awal tahun 1990-an, Microsoft mulai mengembangkan versi baru dari SQL Server untuk dapat berjalan pada *platform* NT. Selama tahap pengembangan ini, Microsoft memutuskan bahwa SQL Server bersifat *tightly coupled* (tingkat interaksinya tinggi) dengan sistem operasi NT.

Pada tahun 1993, Microsoft mengeluarkan Windows NT 3.1 dan SQL Server 4.2 untuk NT. Hal ini menunjukkan bahwa filosofi Microsoft untuk menggabungkan *high-performance database* dengan *interface* yang mudah digunakan benar-benar terbukti. Ini menyebabkan Microsoft dengan segera menjadi vendor software relational *database* yang populer.

Tahun 1994, Microsoft dan Sybase secara resmi mengakhiri kerjasama mereka. Pada tahun 1995, Microsoft mengeluarkan SQL Server 6.0. *Release* ini merupakan penulisan ulang teknologi utama SQL Server dengan peningkatan pada kinerjanya, replikasi *built-in*, dan pengiriman administrasi yang terpusat.

Tahun 1996 Microsoft mengeluarkan SQL Server 6.5. Versi ini membawa peningkatan dan perbaikan terhadap teknologi yang ada dengan menyediakan fitur-fitur baru. Tahun 1997, Microsoft mengeluarkan SQL Server 6.5 Enterprise Edition, SQL Server 7.0 yang merupakan *engine database* yang ditulis ulang secara lengkap dikeluarkan pada tahun 1998.

Tahun 2000, Microsoft mengeluarkan SQL Server 2000. Versi ini dikembangkan berdasarkan SQL Server 7.0 Framework. Menurut tim pengembang SQL Server, perubahan pada *engine database* SQL Server 2000 ini dirancang dan dipersiapkan untuk bertahan sampai dengan 10 tahun kedepan.

Guna memenuhi permintaan pasar terhadap solusi manajemen data, SQL Server selalu berusaha untuk meningkatkan performanya dengan menghadirkan fitur-fitur yang menjadikannya pilihan terbaik sebagai *tools* untuk *system management and tuning* dengan skalabilitas dan reabilitas yang baik.