

BAB 2

LANDASAN TEORI

2.1 Sistem Informasi Manajemen

2.1.1 Pengertian Sistem

Menurut McLeod (2001, p11) system merupakan sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Contoh suatu organisasi atau bidang fungsional cocok untuk menggambarkan ini, dimana organisasi terdiri dari bidang-bidang fungsional yang semuanya mengacu pada tercapainya tujuan organisasi yang telah ditetapkan.

Sistem terdiri dari elemen-elemen yang menunjang terbentuknya system itu sendiri yaitu input, proses transformasi, output. Dimana elemen umpan balik terkadang digunakan untuk menampung informasi dari *output system* dan memberikan kepada sistem sebagai *input* baru.

Sistem ini sendiri terediri dari dua jenis, yaitu sistem terbuka dan sistem tertutup. Suatu sistem yang dihubungkan dengan lingkungannya melalui arus sumber daya disebut sistem terbuka, sedangkan jika sistem tidak lagi dihubungkan dengan lingkungannya maka ini disebut sistem tertutup.

Menurut O'Brien (2003, p8) sistem adalah sebuah kelompok yang terintegrasi dan bekerja sama untuk mencapai tujuan yang sama dengan menerima masukan (*inputs*) dan menghasilkan keluaran (*outputs*) dalam sebuah proses transformasi yang terorganisir dengan baik.

Sistem adalah suatu jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan pokok perusahaan atau mencapai tujuan tertentu dari perusahaan.

2.1.2 Pengertian Sistem Informasi

Menurut O'Brien (2003, p7), sebuah sistem informasi dapat berupa kombinasi teratur dari orang, *hardware*, *software*, jaringan komunikasi dan sumber data yang mengumpulkan, mengubah dan menyebarkan informasi di dalam suatu organisasi.

Menurut Laudon (2003, p7), sistem informasi adalah sebuah kumpulan dari komponen-komponen yang saling berhubungan yang mengumpulkan (atau mengambil kembali), mengolah, menyimpan dan mendistribusikan informasi untuk mendukung pengambilan keputusan, koordinasi dan pengendalian di dalam sebuah organisasi.

Jadi sistem informasi adalah elemen-elemen yang saling berkaitan dengan menggunakan sumber daya untuk mengolah masukan berupa data menjadi keluaran berupa informasi, sehingga berguna bagi pihak yang membutuhkannya.

2.1.3 Pengertian Sistem Informasi Manajemen

Menurut McLeod (2001, p17) sistem informasi manajemen adalah suatu sistem penghasil informasi yang mendukung sekelompok manajer yang mewakili suatu unit organisasi seperti suatu tingkat manajemen atau suatu bidang fungsional.

Menurut O'Brien (2003, p26) sistem informasi manajemen adalah sebuah sistem yang menyediakan informasi dalam bentuk laporan dan gambaran untuk diberikan kepada manajer dan para profesional misalnya analisa penjualan, kinerja produksi, sistem laporan biaya.

2.1.4 *Computer Based Information System (CBIS)*

Menurut Turban et al. (2001, p17), sistem informasi berbasis komputer (CBIS) merupakan sebuah sistem informasi yang menggunakan komputer dan teknologi telekomunikasi untuk melakukan beberapa atau semua tugasnya. Turban et al. (2001, p17) juga menyebutkan beberapa komponen dasar sistem informasi sebagai berikut:

a. *Hardware*

Merupakan satu set perangkat seperti *processor*, *monitor*, *keyboard*, dan *printer* yang dapat menerima, memproses, dan menampilkan data dan informasi.

b. *Software*

Merupakan satu set program komputer yang memungkinkan *hardware* untuk memproses data.

c. *Database*

Merupakan kumpulan dari *file* dan *record* yang dapat menyimpan data dan hubungan diantara *file* dan *record* tersebut.

d. *Network*

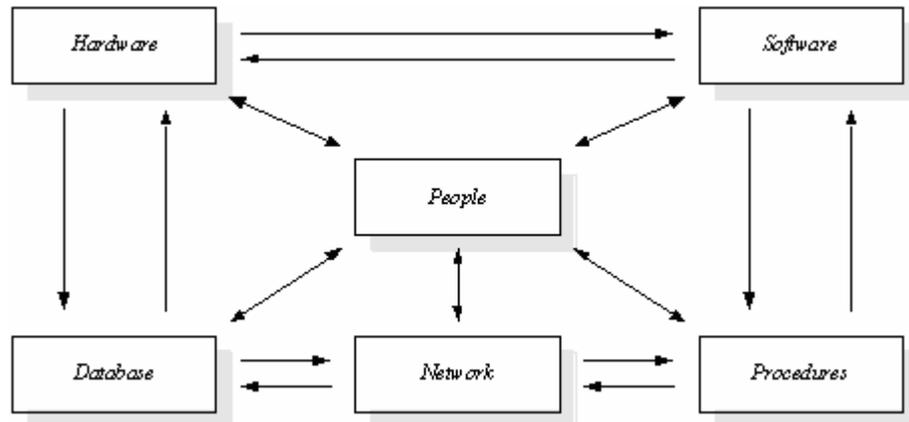
Merupakan sistem penghubung yang memungkinkan pembagian sumber daya diantara beberapa komputer yang berbeda.

e. *Procedures*

Merupakan strategi, kebijakan, metode, dan aturan untuk menggunakan sistem informasi.

f. *People*

Merupakan elemen terpenting dari sistem informasi yang mencakup orang yang bekerja dengan sistem informasi atau menggunakan hasil dari sistem informasi tersebut.



Gambar 2.1 Komponen CBIS

2.2 Manajemen Sumber Daya Manusia

2.2.1 Pengertian Manajemen

Menurut Hasibuan (2002, p1), manajemen adalah ilmu dan seni mengatur proses pemanfaatan sumber daya manusia dan sumber-sumber lainnya secara efektif dan efisien untuk mencapai suatu tujuan tertentu.

Menurut Gomes (2000, p1), manajemen berasal dari kata *to manage* (bahasa inggris), yang artinya mengurus, mengatur, melaksanakan dan mengelola.

2.2.2 Pengertian Sumber Daya Manusia

Sumber daya manusia menurut Gomes (2000, p1) adalah salah satu sumber daya yang ada dalam organisasi, meliputi semua orang yang melakukan aktivitas.

Sumber daya manusia menurut Hasibuan (2002, p244) adalah kemampuan terpadu dari daya pikir dan daya fisik yang dimiliki individu. Perilaku dan sifatnya ditentukan oleh keturunan dan lingkungannya, sedangkan prestasi kerjanya dimotivasi oleh keinginan untuk memenuhi kepuasannya.

2.2.3 Pengertian Manajemen Sumber Daya Manusia

Menurut Dessler (2003, p2), Manajemen sumber daya manusia adalah kebijakan-kebijakan dan praktik yang dibutuhkan seseorang untuk melaksanakan aspek personil atau orang-orang dari pekerjaan manajemennya, termasuk perekrutan, penyaringan, pelatihan, pengimbangan, dan penilaian.

Manajemen sumber daya manusia menurut Hasibuan (2002, p10), merupakan ilmu dan seni mengatur hubungan dan perencanaan tenaga kerja agar efektif dan efisien membantu terwujudnya tujuan perusahaan, dan masyarakat.

Manajemen sumber daya manusia meliputi perencanaan, pengorganisasian, pengawasan, pengembangan dan pemanfaatan sumber daya manusia yang efektif untuk tercapainya berbagai tujuan individu, organisasi, masyarakat, nasional dan internasional.

2.2.4 Tujuan dan Kegiatan Manajemen Sumber Daya Manusia

Menurut Hariandja (2005, pp3-6) tujuan dari manajemen sumber daya manusia yaitu untuk meningkatkan dukungan sumber daya manusia dalam usaha meningkatkan efektivitas organisasi dalam rangka mencapai tujuan. Secara lebih operasional (dalam

arti yang dapat diamati/ diukur) untuk meningkatkan produktivitas pegawai, mengurangi tingkat absensi, mengurangi tingkat perputaran kerja, atau meningkatkan loyalitas para pegawai pada organisasi.

Kegiatan atau aktivitas manajemen sumber daya manusia secara umum dapat dikategorikan menjadi empat, yaitu:

1. Persiapan dan pengadaan

Kegiatan persiapan dan pengadaan meliputi banyak kegiatan, diantaranya adalah kegiatan analisis jabatan, yaitu kegiatan untuk mengetahui jabatan-jabatan yang ada dalam organisasi beserta tugas-tugas yang dilakukan dan persyaratan yang harus dimiliki oleh pemegang jabatan tersebut dan lingkungan kerja di mana aktivitas tersebut dilakukan. Selanjutnya, sebagai landasan kegiatan dilakukan perencanaan sumber daya manusia, yaitu memprediksi dan menentukan kebutuhan tenaga kerja pada masa sekarang dan yang akan datang, baik jumlahnya maupun keahliannya atau jenisnya. Rencana sumber daya manusia akan menunjukkan jumlah yang akan direkrut dan kapan dilakukan rekrutmen untuk menarik calon pegawai yang berpotensi untuk mengisi jabatan. Setelah sekumpulan pelamar diperoleh, dilakukan seleksi untuk mendapatkan pegawai yang memenuhi persyaratan. Kemudian, setelah mereka diterima, sering kali kemampuan mereka sepenuhnya belum sesuai dengan keinginan organisasi, sehingga dilakukanlah program orientasi, setelah itu dilakukan penempatan.

2. Pengembangan dan penilaian

Setelah mereka bekerja secara berkala harus dilakukan pelatihan-pelatihan. Hal ini diperlukan untuk meningkatkan produktivitas pegawai dan menjaga

terjadinya keusangan kemampuan pegawai akibat perubahan-perubahan yang terjadi dalam lingkungan kerja. Kemudian dilakukan penilaian yang bertujuan untuk melihat apakah untuk kerja pegawai sesuai dengan yang diharapkan, dan memberikan umpan balik untuk meningkatkan kemampuan dan kinerja. Selanjutnya membantu perencanaan karir pegawai yang memasuki suatu organisasi senantiasa menginginkan jabatan yang lebih tinggi dan biasanya dengan tanggung jawab dan gaji yang lebih tinggi

3. Pengkompensasian dan perlindungan

Untuk mempertahankan dan memelihara semangat kerja dan motivasi, para pegawai diberi kompensasi dan beberapa kenikmatan atau keuntungan lainnya dalam bentuk program-program kesejahteraan. Hal ini disebabkan pegawai menginginkan balas jasa yang layak sebagai konsekuensi pelaksanaan pekerjaan. Selain itu juga untuk melindungi pegawai dari akibat buruk yang mungkin timbul dari pelaksanaan pekerjaan, serta untuk menjaga kesehatan pegawai.

4. Hubungan-hubungan kepegawaian

Hubungan-hubungan kepegawaian meliputi usaha untuk memotivasi pegawai, memberdayakan pegawai yang dilakukan melalui penataan pekerjaan yang baik, meningkatkan disiplin pegawai agar mematuhi aturan, kebijakan-kebijakan yang ada, dan melakukan bimbingan. Kemudian, bilamana dalam organisasi terbentuk organisasi atau serikat pekerja, organisasi harus melakukan kerja sama yang sinergis, dalam arti saling menguntungkan antara pegawai dan organisasi. Selanjutnya dalam waktu-waktu tertentu harus dilakukan penilaian tentang

sejauh mana manajemen sumber daya manusia tersebut memenuhi fungsinya, yang dilakukannya melalui apa yang disebut audit sumber daya manusia.

2.2.5 Perekrutan

Perekrutan tenaga kerja menurut Sastrohadiwaryo (2002, p138) adalah suatu proses tenaga kerja dan mendorong serta memberikan pengharapan mereka untuk melamar pekerjaan kepada perusahaan.

Seleksi tenaga kerja menurut Sastrohadiwiryoy (2002, p150) adalah kegiatan untuk melakukan dan memilih tenaga kerja yang memenuhi criteria yang telah ditetapkan perusahaan serta memprediksi kemungkinan keberhasilan atau kegagalan individu dalam pekerjaan yang akan diberikan kepadanya.

Tujuan seleksi penerimaan karyawan menurut Hasibuan (2002, p49) adalah untuk mendapatkan :

1. Karyawan yang berkualitas dan potensional.
2. Karyawan yang jujur dan disiplin
3. Karyawan yang cakap dengan penempatan yang benar
4. Karyawan yang terampil dan memiliki gairah dalam bekerja.
5. Karyawan yang dapat memenuhi persyaratan undang-undang perburuhan.
6. Karyawan yang dapat bekerjasama baik secara horizontal dan vertical.
7. Karyawan yang dinamis dan kreatif
8. Karyawan yang inovatif dan bertanggung jawab sepenuhnya.
9. Karyawan yang loyal dan berdedikasi tinggi.
10. Karyawan yang mudah dikembangkan di masa depan.
11. Karyawan yang dapat bekerja secara mandiri

12. Karyawan yang mempunyai perilaku dan budaya malu.

2.2.6 Pelatihan Karyawan

Menurut Sastrohadiwiryono (2002, p200) Pelatihan adalah bagian dari pendidikan yang menyangkut proses belajar untuk memperoleh dan meningkatkan ketrampilan di luar sistem pendidikan yang berlaku dalam waktu yang relatif singkat dan dengan metode yang lebih mengutamakan praktek daripada teori.

Istilah pelatihan sering disamakan dengan istilah pengembangan karena pelatihan mengarahkan untuk mengembangkan potensi karyawan dalam perusahaan. Pengembangan menunjuk kepada kesempatan-kesempatan belajar yang didesain guna membuat perkembangan karyawan.

Menurut Hariandja (2005, p168) Pelatihan dan pengembangan merupakan dua konsep yang sama, yaitu untuk meningkatkan pengetahuan, keterampilan dan kemampuan. Tetapi dilihat dari tujuannya, umumnya kedua konsep tersebut dapat dibedakan. Pelatihan lebih ditekankan pada peningkatan kemampuan untuk melakukan pekerjaan yang spesifik pada saat ini, dan pengembangan lebih ditekankan pada peningkatan pengetahuan untuk melakukan pekerjaan pada masa yang akan datang, yang dilakukan melalui pendekatan yang terintegrasi dengan kegiatan lain untuk mengubah perilaku kerja.

Menurut Siagian (2000, p183), bagi organisasi terdapat paling sedikit tujuh manfaat yang dapat dipetik melalui penyelenggaraan program pelatihan dan pengembangan.

- a) Peningkatan produktivitas kerja organisasi sebagai keseluruhan karena antara lain tidak terjadinya pemborosan, kecermatan melaksanakan tugas, tumbuh suburnya kerjasama antara berbagai kesatuan kerja yang melaksanakan kegiatan yang berbeda dan bahkan spesifik, meningkatnya tekad mencapai sasaran yang telah ditetapkan serta lancarnya koordinasi sehingga organisasi yang bergerak sebagai suatu kesatuan yang utuh.
- b) Terwujudnya hubungan yang serasi antara atasan dan bawahan antara lain karena adanya pendelegasian wewenang, interaksi yang didasarkan pada sikap dewasa baik secara teknikal maupun intelektual, saling menghargai dan adanya kesempatan bagi bawahan untuk berpikir dan bertindak secara inovatif.
- c) Terjadinya proses pengambilan keputusan yang lebih cepat dan tepat karena melibatkan para pegawai yang bertanggung jawab menyelenggarakan kegiatan operasional dan tidak sekedar diperintahkan oleh manajer.
- d) Meningkatkan semangat kerja seluruh karyawan dalam organisasi dengan komitmen organisasional yang tinggi
- e) Mendorong sikap keterbukaan manajemen melalui penerapan gaya manajerial yang partisipatif
- f) Memperlancar jalannya komunikasi yang efektif yang pada gilirannya memperlancar proses perumusan kebijaksanaan organisasi dan operasionalisasinya.
- g) Penyelesaian konflik secara fungsional yang dampaknya adalah tumbuh suburnya rasa persatuan dan suasana kekeluargaan di kalangan para anggota organisasi

Menurut Flippo (Hasibuan, 2002, p70), pelatihan merupakan suatu usaha peningkatan pengetahuan dan keahlian seorang karyawan untuk mengerjakan suatu pekerjaan tertentu.

Menurut Werther dan Keith (Hariandja, 2005, pp170-172) Pelatihan dan pengembangan memberikan manfaat untuk organisasi maupun pegawai itu sendiri. Secara rinci akan dijabarkan dibawah ini:

1. Keuntungan untuk organisasi

Meningkatkan kemampuan organisasi untuk memperoleh keuntungan dan atau sikap yang positif terhadap orientasi keuntungan.

- Memperbaiki pengetahuan dan keterampilan di semua level organisasi.
- Memperbaiki moral tenaga kerja.
- Membantu pegawai mengidentifikasi tujuan organisasi.
- Membantu menciptakan citra perusahaan yang lebih baik.
- Meningkatkan keotentikan, keterbukaan, dan kejujuran.
- Memperbaiki hubungan atasan dan bawahan.
- Memberikan bantuan dalam pengembangan organisasi.
- Membantu menyiapkan pedoman kerja.
- Membantu pemahaman dan pelaksanaan kebijakan.
- Memberikan informasi untuk kebutuhan masa depan di semua bidang organisasi.
- Meningkatkan efektivitas proses pengambilan keputusan dan pemecahan masalah.
- Membantu pengembangan promosi dari dalam.

- Membantu pengembangan keterampilan kepemimpinan, motivasi, loyalitas, sikap yang lebih baik, dan aspek lain yang biasanya ditampilkan oleh pegawai dan manajer yang berhasil.
- Membantu meningkatkan produktivitas dan atau kualitas kerja.
- Membantu menurunkan biaya di beberapa bidang seperti produksi, kepegawaian, administrasi, dan lain-lain.
- Mengembangkan rasa tanggung jawab pada organisasi dan menjadi lebih kompetenserta memiliki kemampuan.
- Memperbaiki hubungan tenaga kerja dan manajemen.
- Mengurangi biaya konsultasi dari luar dengan menggunakan konsultan internen yang kompeten.
- Mendorong terjadinya manajemen pencegahan sebagai kebalikan dari tindakan sesudah terjadi masalah.
- Menghindari perilaku yang tidak sepatutnya seperti menyembunyikan peralatan.
- Menciptakan iklim kerja yang tepat untuk pertumbuhan dan komunikasi organisasi.
- Membantu pegawai menyesuaikan diri dengan perubahan.
- Membantu dalam penanganan konflik, sehingga membantu mencegah stres dan ketegangan.

2. Keuntungan untuk pegawai yang akhirnya akan menguntungkan organisasi.

Membantu pegawai membuat keputusan yang lebih baik dan pemecahan masalah yang efektif melalui pelatihan dan pengembangan factor-faktor motivasi, yaitu

pengakuan, prestasi, pertumbuhan, tanggung jawab, dan perkembangan yang terinternalisasi dan teroperasionalkan.

- Membantu mendorong dan mencapai pengembangan diri sendiri dan kepercayaan diri.
- Membantu seseorang menangani stress, ketegangan, frustrasi, dan konflik.
- Memberikan informasi untuk perbaikan pengetahuan kepemimpinan dan keterampilan dalam berkomunikasi dan bersikap.
- Meningkatkan kepuasan kerja dan pengakuan.
- Memberikan kepada peserta pelatihan sebuah kesempatan untuk berkembang dan pandangan tentang masa depan sendiri.
- Menggerakkan seseorang ke arah tujuan-tujuan pribadi ketika keterampilan interaksi meningkat.
- Memuaskan kebutuhan para pelatih dan juga peserta pelatihan
- Meningkatkan keinginan belajar.
- Membantu seseorang meningkatkan keterampilan berbicara dan mendengar, juga kemampuan menulis bilamana pelatihan-pelatihan dilakukan.
- Membantu menghilangkan ketakutan dalam mencoba tugas baru.

3. Keuntungan dalam hubungan kepegawaian, hubungan-hubungan antar kelompok dan dalam kelompok, dan pelaksanaan kebijaksanaan.

- Memperbaiki komunikasi antar kelompok dan antar individu.
- Memberikan bantuan dalam orientasi pegawai baru dan mereka yang mengambil tugas baru melalui transfer dan promosi

- Memberikan informasi dengan kesempatan yang sama dan tindakan penyesuaian.
- Memberikan informasi mengenai peraturan pemerintah lain dan kebijakan-kebijakan administrasi.
- Meningkatkan keterampilan hubungan antar pribadi.
- Membuat kebijakan organisasi aturan sesuai dengan situasi.
- Memperbaiki moral.
- Memberikan iklim yang baik untuk belajar, berkembang, dan berkoordinasi.
- Membuat organisasi tempat yang lebih baik untuk bekerja dan hidup.

2.2.7 Penilaian Kinerja

Menurut Sastrohadiwiryono (2002, p231) mengemukakan bahwa penilaian kinerja adalah suatu kegiatan yang dilakukan manajemen atau penyelia untuk menilai kinerja tenaga kerja dengan cara membandingkan kinerja mereka atas deskripsi pekerjaan dalam suatu periode tertentu biasanya setiap akhir tahun.

Hasibuan (2002, p87) berpendapat bahwa penilaian prestasi kerja adalah menilai rasio hasil kerja nyata dengan standar kualitas maupun kuantitas yang dihasilkan setiap karyawan.

Dari definisi di atas dapat disimpulkan bahwa penilaian kinerja karyawan adalah suatu proses pengukuran keaktifan karyawan dalam menyelesaikan pekerjaannya atau tugasnya dengan tujuan menghasilkan suatu susunan peringkat atau memberikan suatu gambaran mengenai prestasi kerja seorang karyawan.

Tidak terdapat kesamaan antara perusahaan yang satu dengan yang lain dalam menentukan unsur yang harus dinilai, tetapi pada umumnya unsur-unsur yang perlu dinilai dalam proses penilaian kinerja adalah:

a. Kesetiaan

Kesetiaan yang dimaksud adalah tekad dan kesanggupan menaati, melaksanakan, dan mengamalkan sesuatu yang ditaati dengan penuh kesadaran dan tanggung jawab.

b. Prestasi kerja

Kinerja yang dicapai seseorang tenaga kerja dalam melaksanakan tugas dan pekerjaan yang diberikan kepadanya.

c. Tanggung jawab

Adalah kesanggupan seorang tenaga kerja dalam menyelesaikan tugas dan pekerjaan yang diberikan kepadanya dengan baik dan tepat waktu serta berani memikul resiko atas keputusan yang telah diambilnya.

d. Ketaatan

Kesanggupan seorang tenaga kerja untuk mentaati segala ketentuan, peraturan perundang-undangan dan peraturan kedinasan yang berlaku, mentaati perintah kedinasan yang diberikan atasan yang berwenang, serta kesanggupan untuk tidak melanggar larangan yang telah ditetapkan perusahaan maupun pemerintah, baik tertulis maupun tidak tertulis.

e. Kejujuran

Karyawan itu harus jujur terutama pada dirinya sendiri, bawahan, perjanjian-perjanjian dalam menjalankan atau mengelola jabatan tersebut, harus sesuai dengan antara kata dengan tindakan atau perbuatan.

f. Kerja sama

Adalah kemampuan seorang tenaga kerja untuk bekerja bersama dengan orang lain dalam menyelesaikan suatu tugas dan pekerjaan yang telah ditetapkan, sehingga mencapai daya guna dan hasil guna yang sebesar-besarnya.

g. Prakarsa

Kemampuan seorang tenaga kerja untuk mengambil keputusan, langkah-langkah atau melaksanakan suatu tindakan yang diperlukan dalam melaksanakan tugas pokok tanpa menunggu perintah dan bimbingan manajemen lininya.

h. Kepemimpinan

Merupakan kemampuan yang dimiliki seorang tenaga kerja untuk meyakinkan orang lain sehingga dapat dikerahkan secara maksimum untuk melaksanakan tugas pokok. Penilaian unsur kepemimpinan bagi tenaga kerja sebenarnya khusus untuk tenaga kerja yang memiliki jabatan hierarki dalam perusahaan.

2.3 Konsep Analisis & Perancangan Sistem Informasi Berbasis Objek

2.3.1 *Object Oriented Analysis and Design (OOAD)*

Object-Oriented Analysis and Design (OOAD) adalah metode untuk menganalisa dan merancang sistem dengan pendekatan berorientasi *object* (Mathiassen et al, 2000, p135). *Object* diartikan sebagai suatu entitas yang memiliki identitas, *state* dan *behavior* (mathiassen et al, 2000, p4). Pada analisa, identitas sebuah *object* menjelaskan bagaimana seorang user membedakannya dari *object* lain, dan *behavior object* digambarkan melalui event yang dilakukannya. Sedangkan pada perancangan, identitas sebuah *object* digambarkan dengan cara bagaimana *object* lain mengenalinya

sehingga dapat diakses, dan *behavior object* digambarkan dengan *operation* yang dapat dilakukan *object* tersebut yang dapat mempengaruhi *object* lain dalam sistem.

2.3.2 Objek dan Class

Objek merupakan sebuah entitas yang memiliki identitas, status, dan perilaku (Mathiassen et al., 2000,p4). Contoh dari objek misalnya pelanggan yang merupakan entitas dengan identitas yang spesifik, dan memiliki status dan perilaku tertentu yang berbeda antara satu pelanggan dengan pelanggan yang lain. Sedangkan *class* merupakan deskripsi dari kumpulan objek yang memiliki struktur, pola perilaku, dan atribut yang sama (Mathiassen et al., 2000,p4). Untuk dapat lebih memahami objek, biasanya objek-objek tersebut sering digambarkan dalam bentuk *class*.

2.3.3 Konsep *Oriented Analysis and Design* (OOAD)

Terdapat tiga buah konsep atau teknik dasar dalam proses analisa dan perancangan berorientasi objek, yaitu:

1. *Encapsulation*

Encapsulation dalam bahasa pemrograman berorientasi objek secara sederhana berarti pengelompokan fungsi. Pengelompokan ini bertujuan agar *developer* tidak perlu membuat *coding* untuk fungsi yang sama, melainkan hanya perlu memanggil fungsi yang telah dibuat sebelumnya.

2. *Inheritance*

Inheritance dalam bahasa pemrograman berorientasi objek secara sederhana berarti menciptakan sebuah *class* baru yang memiliki sifat-sifat dan

karakteristik-karakteristik sama dengan yang dimiliki *class* induknya disamping sifat-sifat dan karakteristik-karakteristik individualnya.

3. *Polymorphism*

Polymorphism berarti kemampuan dari tipe objek yang berbeda untuk menyediakan atribut dan operasi yang sama dalam hal yang berbeda.

Polymorphism adalah hasil natural dari fakta bahwa objek dari tipe yang berbeda atau bahkan dari sub-tipe yang berbeda dapat menggunakan atribut dan operasi yang sama.

2.3.4 Keuntungan dan Kelemahan *Object Oriented Analysis and Design (OOAD)*

Mathiassen et al. (2000, pp5-6) menyebutkan bahwa terdapat keuntungan menggunakan OOAD diantaranya adalah:

1. OOAD memberikan informasi yang jelas mengenai *context* sistem.
2. Dapat menangani data yang seragam dalam jumlah yang besar dan mendistribusikannya ke seluruh bagian organisasi.
3. Berhubungan erat dengan analisa berorientasi objek, perancangan berorientasi objek, *user interface* berorientasi objek, dan pemrograman berorientasi objek.

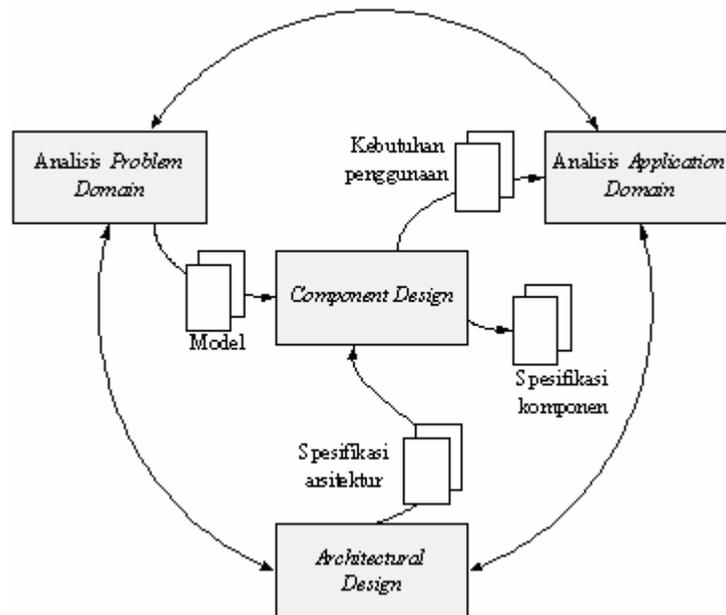
Selain keuntungan yang diperoleh dalam menggunakan OOAD seperti yang telah disebutkan di atas, ternyata juga terdapat beberapa kelemahan yang berhasil diidentifikasi oleh McLeod (2001, p615) yaitu:

1. Diperlukan waktu lama untuk memperoleh pengalaman pengembangan.
2. Kesulitan metodologi untuk menjelaskan sistem bisnis yang rumit.

3. Kurangnya pilihan peralatan pengembangan yang khusus disesuaikan untuk sistem bisnis.

2.3.5 Aktivitas Utama *Object Oriented Analysis and Design (OOAD)*

Mathiassen et al. (2000, pp14-15) menjelaskan empat buah aktivitas utama dalam analisa dan perancangan berorientasi objek yang digambarkan dalam Gambar 2.2 berikut ini.



Gambar 2.2 Aktivitas Utama dalam OOAD

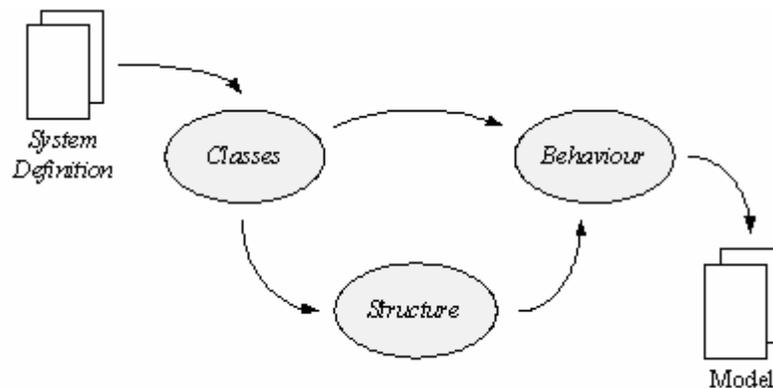
Berikut ini merupakan penjelasan lebih rinci mengenai keempat aktivitas utama dalam melakukan analisa dan perancangan berorientasi objek menurut Mathiassen et al. (2000, pp14-15):

1. *Analisis Problem Domain*

Problem domain merupakan bagian dari situasi yang diatur, diawasi, dan dikendalikan oleh sistem. Tujuan melakukan analisis *problem domain* adalah mengidentifikasi dan memodelkan *problem domain*. Analisis *problem*

domain terbagi menjadi tiga aktivitas yang digambarkan dalam Gambar 3.10, yaitu:

- a. Memilih objek, *class*, dan *event* yang akan menjadi elemen model *problem domain*.
- b. Membangun model dengan memusatkan perhatian pada relasi struktural antara *class* dan objek.
- c. Mendeskripsikan properti dinamis dan atribut untuk setiap *class*.



Gambar 2.3 Aktivitas Analisis *Problem Domain*

Pada aktivitas *classes*, langkah awal yang perlu dilakukan adalah menentukan *class*. Langkah berikutnya adalah membuat sebuah *event table* yang dapat membantu menentukan *event-event* yang dimiliki oleh setiap.

Pada aktivitas *structure*, *class-class* yang telah ditentukan sebelumnya akan dihubungkan berdasarkan tiga jenis hubungan yaitu generalisasi, agregasi, atau asosiasi sehingga menjadi sebuah skema yang disebut *class diagram*.

Dalam aktivitas *behavior*, definisi *class* dalam *class diagram* akan diperluas dengan menambahkan deskripsi pola perilaku dan atribut dari masing-masing *class*. Pola perilaku dari *class* terdiri dari tiga jenis, yaitu:

- *Sequence*

Merupakan *event* yang terjadi secara berurutan satu per satu.

- *Selection*

Merupakan pemilihan salah satu dari beberapa *event* yang terjadi.

- *Iteration*

Merupakan *event* yang terjadi berulang kali.

Hasil dari aktivitas ini adalah sebuah *statechart diagram* yang menunjukkan perubahan status dari masing-masing *class* yang dikarenakan oleh *event* tertentu mulai dari *initial state* sampai dengan *final state*.

2. Analisis *Application Domain*

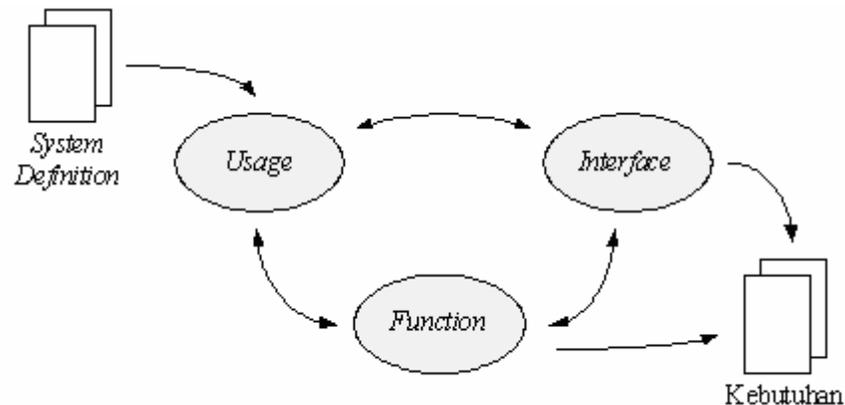
Menurut Mathiassen, et al (2000, p115) *application-domain* adalah organisasi yang mengatur, memonitor atau mengendalikan *problem-domain*.

Analisis *application-domain* memfokuskan bagaimana target dalam sistem akan digunakan dengan menentukan *function* dan *interface* sistem. Sama seperti analisis *problem domain*, analisis *application domain* juga terdiri dari beberapa aktivitas antara lain:

- a. Menentukan penggunaan sistem dan bagaimana sistem berinteraksi dengan *user*.

- b. Menentukan fungsi dan kemampuan sistem dalam mengolah informasi.
- c. Menentukan kebutuhan *interface* sistem dan merancang *interface*.

Berikut ini merupakan gambaran aktivitas-aktivitas yang dilakukan pada saat melakukan analisis *application domain*.



Gambar 2.4 Aktivitas Analisis *Application Domain*

- ***Usage***

Menurut Mathiassen, et al (2000, p119-120) kegiatan usage adalah kegiatan pertama dalam analisis *application-domain* yang bertujuan untuk menentukan bagaimana aktor-aktor yang merupakan pengguna atau sistem yang berinteraksi dengan sistem yang dituju. Interaksi antara aktor dengan sistem tersebut dinyatakan dalam *use case diagram*.

Use case dapat dimulai oleh aktor atau oleh sistem target. Hasil dari analisis kegiatan *usage* ini adalah deskripsi lengkap dari semua *use case* dan aktor yang ada yang digambarkan dalam tabel aktor atau *use case diagram*. Cara untuk mengidentifikasi aktor adalah mengetahui alasan aktor menggunakan sistem. Masing-masing aktor memiliki alasan yang berbeda untuk menggunakan sistem. Cara lainnya yaitu dengan melihat

peran dari aktor seperti yang dinyatakan oleh *use case* dimana aktor tersebut terlibat. Masing-masing aktor memiliki peran yang berbeda-beda.

Use case dapat digambarkan dengan menggunakan spesifikasi *use case*, dimana *use case* dijelaskan secara singkat namun jelas dan dapat disertai dengan keterangan objek sistem yang terlibat dan *function* dari *use case* tersebut atau dengan diagram *statechart* karena *use case* adalah sebuah fenomena yang dinamik

- ***Function***

Menurut Mahiassen, et al (2000, p137-138). *Function* memfokuskan pada bagaimana cara sebuah sistem dapat membantu aktor dalam melaksanakan pekerjaan mereka. *Function* memiliki empat tipe yang berbeda, yaitu:

1. ***Update***

Fungsi *update* diaktifkan oleh *event problem domain* dan menghasilkan perubahan status model.

2. ***Signal***

Fungsi *signal* diaktifkan oleh perubahan status model dan menghasilkan reaksi di dalam *context*.

3. ***Read***

Fungsi *read* diaktifkan oleh kebutuhan *actor* akan informasi dan menghasilkan tampilan model sistem yang relevan.

4. *Compute*

Fungsi *compute* diaktifkan oleh kebutuhan *actor* akan informasi dan berisi perhitungan yang dilakukan baik oleh *actor* maupun oleh model. Hasilnya adalah tampilan dari hasil perhitungan yang dilakukan.

Tujuan dari kegiatan *function* adalah untuk menentukan kemampuan sistem memproses informasi. Hasil dari kegiatan ini adalah sebuah daftar *function-function* yang merinci *function-function* yang kompleks. Daftar *function* harus lengkap menyatakan secara keseluruhan kebutuhan kolektif dari pelanggan dan aktor sehingga harus konsisten dengan *use case*.

Cara untuk mengidentifikasi *function* adalah dengan melihat deskripsi *problem domain* yang dinyatakan dalam kelas dan *event*, dan melihat deskripsi *application domain* yang dinyatakan dalam *use case*. Kelas dapat menyebabkan munculnya kebutuhan terhadap *function update*, sementara *usecase* dapat menyebabkan munculnya segala macam tipe *function*.

- ***User Interface***

Menurut Mahiassen, et al (2000, p151-152). *Interface* menghubungkan sistem dengan semua aktor yang berhubungan dalam konteks. Ada dua jenis *interface*, yaitu: *interface* pengguna yang menghubungkan pengguna dengan sistem dan *interface* sistem yang menghubungkan sistem dengan sistem lainnya.

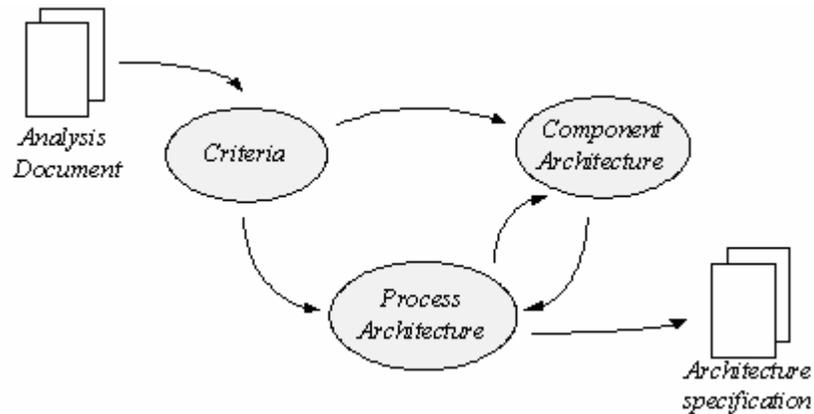
Sebuah *user interface* yang baik harus dapat beradaptasi dengan pekerjaan dan pemahaman *user* terhadap sistem. Kualitas *interface* pengguna ditentukan oleh kegunaan atau *usability interface* tersebut bagi pengguna. *Usability* bergantung pada siapa yang menggunakan dan situasi pada saat sistem tersebut digunakan. Oleh sebab itu, *usability* bukan sebuah ukuran yang pasti dan objektif.

Kegiatan analisis *user interface* ini berdasarkan pada hasil dari kegiatan analisis lainnya, seperti model *problem domain*, kebutuhan *functional* dan *use case*. Hasil dari kegiatan ini adalah sebuah deskripsi elemen-elemen *interface* pengguna dan *interface* sistem yang lengkap, dimana kelengkapan menunjukkan pemenuhan kebutuhan pengguna. Hasil ini harus dilengkapi dengan sebuah diagram navigasi yang menyediakan sebuah ringkasan dari elemen-elemen *user interface* dan perubahan antara elemen-elemen tersebut (p159).

3. *Architectural Design*

Architectural design berfungsi sebagai kerangka kerja dalam aktivitas pengembangan sistem dan menghasilkan struktur komponen dan proses sistem. Tujuannya adalah untuk menstrukturisasi sebuah sistem yang terkomputerisasi.

Tahap *architectural design* terdiri dari tiga aktivitas yaitu *criteria*, *component architecture*, dan *process architecture* seperti yang digambarkan pada Gambar 2.5



Gambar 2.5 Aktivitas *Architectural Design*

Criterion merupakan properti yang diinginkan dari sebuah arsitektur.

Tabel 2.1 menunjukkan *criterion* yang telah ditentukan oleh para peneliti untuk menentukan kualitas dari sebuah *software*.

Tabel 2.1 *Criteria* untuk Menentukan Kualitas *Software*

<i>Criterion</i>	Ukuran
<i>Usable</i>	Kemampuan sistem beradaptasi dengan <i>context</i> organisasional dan teknikal.
<i>Secure</i>	Pencegahan akses ilegal terhadap data dan fasilitas.
<i>Efficient</i>	Eksplorasi ekonomis dari fasilitas <i>technical platform</i> .
<i>Correct</i>	Kesesuaian dengan kebutuhan.
<i>Reliable</i>	Fungsi yang dijalankan secara tepat.
<i>Maintainable</i>	Biaya untuk mencari dan memperbaiki kerusakan sistem.
<i>Testable</i>	Biaya untuk menjamin bahwa sistem melakukan fungsinya.
<i>Flexible</i>	Biaya memodifikasi sistem.
<i>Comprehensible</i>	Usaha yang diperlukan untuk memahami sistem.
<i>Reusable</i>	Penggunaan bagian dari sistem ke dalam sistem lain yang berkaitan.
<i>Portable</i>	Biaya memindahkan sistem ke <i>technical platform</i> lain.
<i>Interoperable</i>	Biaya pemasangan sistem dengan sistem lain.

Mathiassen et al. (2000, pp179-182) menyebutkan bahwa kriteria *usable*, *flexible*, dan *comprehensible* tergolong sebagai kriteria umum yang harus dimiliki oleh sebuah sistem dan menentukan baik tidaknya suatu rancangan sistem.

Component architecture adalah struktur sistem dari komponen-komponen yang berkaitan. Dalam aktivitas ini, perlu ditentukan pola arsitektural yang paling sesuai dengan model sistem. Pola-pola arsitektural tersebut antara lain:

- *Layered Architecture Pattern*
- *Generic Architecture Pattern*
- *Client-Server Architecture Pattern*

Hasil dari aktivitas ini adalah sebuah *component diagram* yang merupakan *class diagram* yang dilengkapi dengan spesifikasi komponen yang kompleks.

Process architecture adalah sebuah struktur eksekusi sistem yang terdiri dari proses-proses yang saling tergantung satu sama lain. Dalam aktivitas ini juga perlu menentukan pola distribusi yang sesuai dengan model sistem.

Pola-pola distribusi yang ada antara lain:

- *Centralized Pattern*
- *Distributed Pattern*
- *Decentralized Pattern*

Hasil dari aktivitas ini adalah sebuah *deployment diagram* yang menunjukkan *processor* dengan komponen program dan *active objects*.

4. *Component Design*

Menurut Mathiassen, et al. (2000, p231) *Component design* bertujuan untuk menentukan implementasi kebutuhan di dalam kerangka kerja arsitektural. Kegiatan *component design* bermula dari spesifikasi arsitektural dan kebutuhan sistem. Hasilnya adalah deskripsi mengenai komponen-komponen yang saling berhubungan dengan sistem. *Component design* terdiri dari tiga aktivitas, yaitu:

a. *Model component*

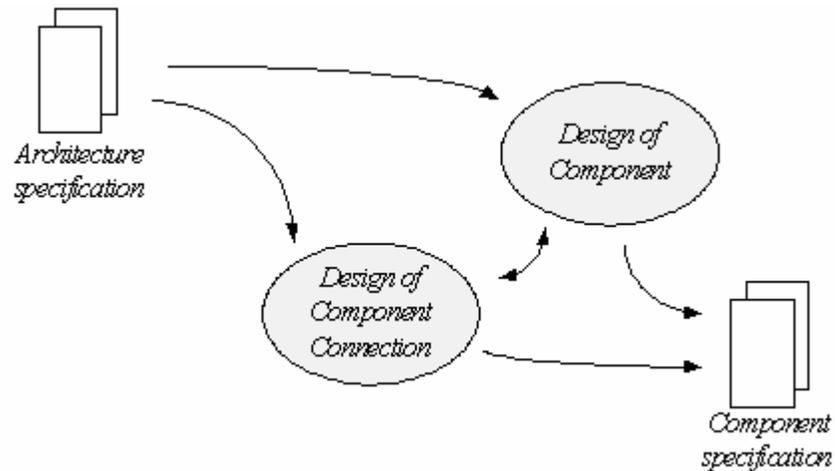
Menurut Mathiassen, et al (2000, p235) *Model component* adalah bagian dari sistem yang mengimplementasikan model *problem domain*. Konsep utama dalam desain komponen model adalah struktur. Dalam aktivitas ini dihasilkan sebuah *class diagram* yang telah direvisi.

b. *Function component*

Menurut Mathiassen, et al (2000, p251) komponen *function* adalah bagian dari sistem yang mengimplementasikan kebutuhan fungsional. Tujuan dari *function* komponen adalah memberikan akses bagi *usr interface* dan komponen sistem lainnya ke model.

c. *Connecting component*

Merupakan desain hubungan antar komponen untuk memperoleh rancangan yang fleksibel dan mudah dimengerti. Hasilnya adalah *class diagram* yang berhubungan dengan komponen-komponen sistem. Gambar 2.6 berikut ini menggambarkan aktivitas-aktivitas yang terdapat dalam *component design*.



Gambar 2.6 Aktivitas *Component Design*

2.4 *Unified Modeling Language (UML)*

2.4.1 Sejarah UML

Pada akhir tahun 80-an dan awal tahun 90-an, sudah banyak terdapat metode pemodelan berorientasi objek yang digunakan pada industri-industri, diantaranya *Booch Method*, *Object Modeling Technique (OMT)* yang diperkenalkan oleh James Rumbaugh, dan *Object-Oriented Software Engineering (OOSE)* yang diperkenalkan oleh Ivar Jacobson. Keberadaan berbagai metode tersebut justru menjadi masalah utama dalam pengembangan sistem berorientasi objek, karena dengan banyaknya metode pemodelan objek yang digunakan akan membatasi kemampuan untuk berbagi model antar proyek dan antar tim pengembang. Hal tersebut disebabkan oleh berbedanya konsep masing-masing metode pemodelan objek sehingga menghambat komunikasi antara anggota tim dengan *user* yang berujung pada banyaknya kesalahan atau *error* pada proyek. Dikarenakan masalah-masalah tersebut, maka diperlukanlah suatu standarisasi penggunaan bahasa pemodelan.

Pada tahun 1994, Grady Booch dan James Rumbaugh bekerja sama dan menyatukan metode pengembangan berorientasi objek mereka dengan tujuan untuk menciptakan sebuah sistem pengembangan berorientasi objek yang standar. Pada tahun 1995 Ivar Jacobson ikut bergabung dengan mereka dan ketiganya memusatkan perhatian untuk menciptakan sebuah bahasa pemodelan objek yang standar, bukan lagi berkonsentrasi pada metode atau pendekatan berorientasi objek. Berdasarkan pemikiran ketiga tokoh tersebut, maka akhirnya pada tahun 1997 bahasa pemodelan objek standar *Unified Modeling Language* (UML) versi 1.0 mulai diperkenalkan kepada masyarakat luas.

UML bukan merupakan metode untuk mengembangkan sistem, melainkan hanya berupa notasi yang kemudian pada saat ini diterima dengan luas sebagai bahasa pemodelan objek yang standar. *Object Management Group* (OMG) mengadopsi UML pada bulan November 1997 dan sejak saat itu terus mengembangkannya berdasarkan pada kebutuhan dunia industri. Pada tahun 2004, telah diluncurkan UML versi 1.4 dan pada saat itu juga OMG telah mulai merencanakan pengembangan UML versi 2.0.

2.4.2 Implementasi

Menurut Mathiassen et al (2000, p313) analisis dan desain *object oriented* diikuti dengan implementasi, dan pendekatannya akan bergantung pada ketersediaannya akan *computer-based tools*. Tujuannya adalah untuk mengimplementasikan desain sistem dalam sebuah *technical platform*, dan hasil dari implementasi adalah sebuah kumpulan dari bagian-bagian *software* yang mengimplementasikan sebuah desain *object oriented*.

2.4.3 Notasi UML

Notasi (Mathiassen et al, 2000, p237) adalah bahasa *textual* dan *graphical* untuk menggambarkan sebuah sistem dan konteksnya yang diformalisasikan secara terpisah. Tujuannya adalah untuk menyederhanakan komunikasi dan dokumentasi. Notasi UML (*Unified Modelling Language*) yang digunakan, yaitu:

2.4.3.1 Class Diagram

Class Diagram menggambarkan struktur objek dari sistem. *Class diagram* menunjukkan *class* objek yang membentuk sistem dan hubungan struktural diantara *class* objek tersebut (Mathiassen et al., 2000, p336). Terdapat tiga jenis hubungan antar *class* yang biasa digunakan dalam *class diagram* (Whitten et al., 2004, pp455-459). Ketiga jenis hubungan tersebut antara lain:

1. Asosiasi

Asosiasi merupakan hubungan statis antar dua objek atau *class*. Hubungan ini menggambarkan apa yang perlu diketahui oleh sebuah *class* mengenai *class* lainnya. Hubungan ini memungkinkan sebuah objek atau *class* mereferensikan objek atau *class* lain dan saling mengirimkan pesan.

2. Generalisasi (atau Spesialisasi)

Dalam hubungan generalisasi, terdapat dua jenis *class*, yaitu *class supertype* dan *class subtype*. *Class supertype* atau *class induk* memiliki atribut dan *behavior* yang umum dari hirarki tersebut. *Class subtype* atau *class anak* memiliki atribut dan *behavior* yang unik dan juga memiliki atribut dan *behavior* milik *class* induknya. *Class induk*

merupakan generalisasi dari *class* anaknya, sedangkan *class* anak merupakan spesialisasi dari *class* induknya.

3. Agregasi

Agregasi merupakan hubungan yang unik dimana sebuah objek merupakan bagian dari objek lain. Hubungan agregasi tidak simetris dimana jika objek B merupakan bagian dari objek A, namun objek A bukan merupakan bagian dari objek B. Pada hubungan ini, objek yang menjadi bagian dari objek tertentu tidak akan memiliki atribut atau *behavior* dari objek tersebut.

2.4.3.2 Statechart Diagram

Statechart Diagram digunakan untuk memodelkan perilaku dinamis dari sebuah objek dalam sebuah *class* yang spesifik dan berisi *state* dan *transition* (Mathiassen et al., 2000, p341). *Statechart diagram* mengilustrasikan siklus objek hidup yaitu berbagai status yang dapat dimiliki objek dan *event* yang menyebabkan status objek berubah menjadi status lain (Whitten et al., 2004, p700).

Statechart diagram dibuat dengan langkah-langkah sebagai berikut (Whitten et al., 2004, p700):

1. Mengidentifikasi *initial* dan *final state*.
2. Mengidentifikasi status objek selama masa hidup objek tersebut.
3. Mengidentifikasi *event* pemicu perubahan status objek.
4. Mengidentifikasi jalur perubahan status.

2.4.3.3 Use Case Diagram

Use Case Diagram menggambarkan interaksi antara sistem dan *user* (Whitten et al., 2004, p441). *Use case diagram* mendeskripsikan secara grafis hubungan antara *actors* dan *use case* (Mathiassen et al., 2000, p343). Penjelasan *use case* biasa ditambahkan untuk menjelaskan langkah-langkah interaksi.

2.4.3.4 Sequence Diagram

Bennet et al. (2006, p253) mengemukakan bahwa *sequence diagram* menunjukkan interaksi antar objek yang diatur berdasarkan urutan waktu. *Sequence diagram* dapat digambarkan dalam berbagai *level of detail* yang berbeda untuk memenuhi tujuan yang berbeda-beda pula dalam daur hidup pengembangan sistem. Aplikasi *sequence diagram* yang paling umum adalah untuk menggambarkan interaksi antar objek yang terjadi pada sebuah *use case* atau sebuah *operation*.

Bennet et al. (2006, pp253-254) menyatakan bahwa setiap *sequence diagram* harus diberikan *frame* yang memiliki *heading* dengan menggunakan notasi **sd** yang merupakan kependekan dari *sequence diagram*. Bennet et al. (2006, p270) juga menyatakan bahwa terdapat beberapa notasi penulisan *heading* pada setiap *frame* yang terdapat dalam *sequence diagram*, antara lain:

a. **alt**

Notasi **alt** merupakan kependekan dari *alternatives* yang menyatakan bahwa terdapat beberapa buah alternatif jalur eksekusi untuk dijalankan.

b. **opt**

Notasi **opt** merupakan kependekan dari *optional* dimana *frame* yang memiliki *heading* ini memiliki status pilihan yang akan dijalankan jika syarat tertentu dipenuhi.

c. **loop**

Notasi **loop** menyatakan bahwa *operation* yang terdapat dalam *frame* tersebut dijalankan secara berulang selama kondisi tertentu.

d. **break**

Notasi **break** mengindikasikan bahwa semua *operation* yang berada setelah *frame* tersebut tidak dijalankan.

e. **par**

Merupakan kependekan dari *parallel* yang mengindikasikan bahwa *operation* dalam *frame* tersebut dijalankan secara bersamaan.

f. **seq**

Notasi **seq** merupakan kependekan dari *weak sequencing* yang berarti *operation* yang berasal dari *lifeline* yang berbeda dapat terjadi pada urutan manapun.

g. **strict**

Notasi **strict** merupakan kependekan dari *strict sequencing* yang menyatakan bahwa *operation* harus dilakukan secara berurutan.

h. **neg**

Notasi **neg** merupakan kependekan dari *negative* yang mendeskripsikan operasi yang tidak *valid*.

i. **critical**

Frame yang memiliki heading **critical** menyatakan bahwa operasi-operasi yang terdapat di dalamnya tidak memiliki sela yang kosong.

j. **ignore**

Notasi ini mengindikasikan bahwa tipe pesan atau parameter yang dikirimkan dapat diabaikan dalam interaksi.

k. **consider**

Consider menyatakan pesan mana yang harus dipertimbangkan dalam interaksi.

l. **assert**

Merupakan kependekan dari *assertion* yang menyatakan urutan pesan yang *valid*.

m. **ref**

Notasi **ref** merupakan kependekan dari *refer* yang menyatakan bahwa *frame* mereferensikan *operation* yang terdapat di dalamnya pada sebuah *sequence diagram* tertentu.

2.4.3.5 *Navigation Diagram*

Navigation Diagram merupakan *statechart diagram* khusus yang berfokus pada *user interface* (Mathiassen et al., 2000, p344). Diagram ini menunjukkan *window-window* dan transisi diantara *window-window* tersebut.

Sebuah *window* dapat digambarkan sebagai sebuah *state*. *State* ini memiliki nama dan berisi gambar miniatur *window*. Transisi antar *state* dipicu oleh ditekannya sebuah tombol yang menghubungkan dua *window*.

2.4.3.6 *Component Diagram*

Component Diagram merupakan diagram implementasi yang digunakan untuk menggambarkan arsitektur fisik dari *software* sistem. Diagram ini dapat menunjukkan bagaimana *coding* pemrograman terbagi menjadi komponen-komponen dan juga menunjukkan ketergantungan antar komponen tersebut (Whitten et al., 2004, p442).

Sebuah komponen digambarkan dalam UML sebagai sebuah kotak dengan dua kotak kecil di sebelah kirinya. Ketergantungan antar dua komponen menunjukkan bagaimana kedua komponen tersebut saling berkomunikasi.

2.4.3.7 *Deployment Diagram*

Deployment Diagram, sama seperti *component diagram*, juga merupakan diagram implementasi yang menggambarkan arsitektur fisik sistem. Perbedaannya, *deployment diagram* tidak hanya menggambarkan arsitektur fisik *software* saja, melainkan *software* dan *hardware*. Diagram ini menggambarkan komponen *software*, *processor*, dan peralatan lain yang melengkapi arsitektur sistem (Whitten et al., 2004, p442). Menurut Mathiassen et al. (2000, p340), *deployment diagram* menunjukkan konfigurasi sistem dalam bentuk *processor* dan objek yang terhubung dengan *processor* tersebut.

Setiap kotak dalam *deployment diagram* menggambarkan sebuah *node* yang menunjukkan sebuah *hardware*. *Hardware* dapat berupa PC, *mainframe*, *printer*, atau bahkan *sensor*. *Software* yang terdapat di dalam *node* digambarkan dengan simbol komponen. Garis yang menghubungkan *node* menunjukkan jalur komunikasi antar *device*.

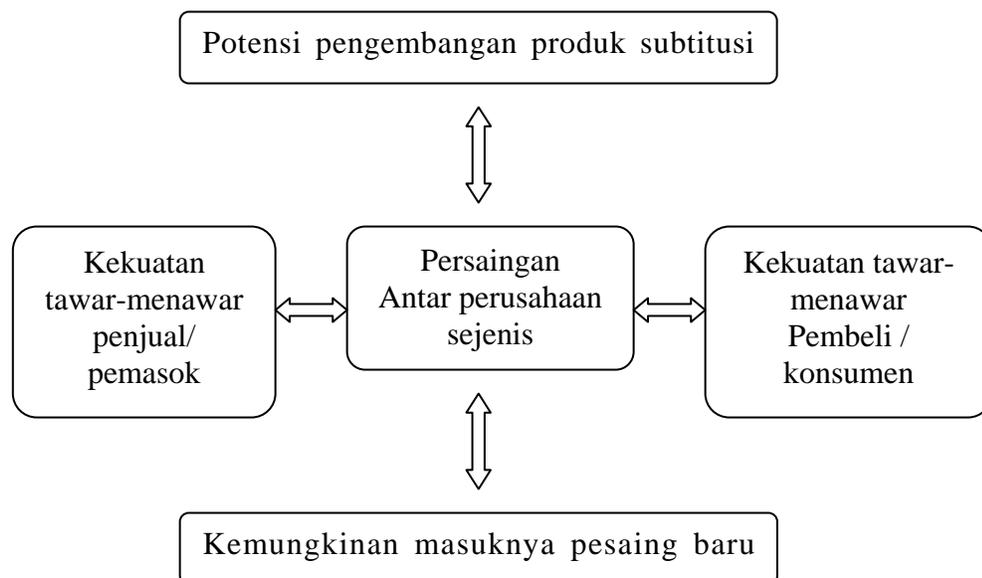
2.5 Metode Analisis Bisnis

2.5.1 Lima Kekuatan Porter

Menurut David (2006, p130-135) hakikat persaingan suatu industri dapat dilihat sebagai kombinasi atas lima kekuatan:

1. Persaingan antar perusahaan sejenis
2. Kemungkinan masuknya pesaing baru.
3. Potensi pengembangan produk substitusi.
4. Kekuatan tawar-manawar penjual/pemasok
5. Kekuatan tawar manawar pembeli/konsumen.

Kelima kekuatan persaingan ini dapat dilihat pada gambar di bawah ini. Berikut ini akan dibahas secara singkat kelima kekuatan persaingan tersebut :



Gambar 2.7 Model Lima Kekuatan Porter
Sumber : David (2006, p131)

1. Persaingan di antara perusahaan sejenis

Persaingan antar perusahaan sejenis biasanya merupakan kekuatan terbesar dalam lima kekuatan kompetitif. Strategi yang dijalankan oleh suatu perusahaan dapat berhasil hanya jika mereka memberikan keunggulan kompetitif dibanding strategi yang dijalankan perusahaan pesaing. Perubahan strategi oleh satu perusahaan mungkin akan mendapat serangan balasan, seperti menurunkan harga, meningkatkan kualitas, menambah *feature*, menyediakan jasa, memperpanjang garansi, dan meningkatkan iklan.

Intensitas persaingan di antara perusahaan sejenis yang bersaing cenderung meningkat karena jumlah pesaing semakin bertambah, karena pesaing semakin seragam dalam hal ukuran dan kemampuan, karena permintaan untuk produk industri menurun, dan karena pemotongan harga menjadi semakin umum.

Persaingan juga meningkat ketika pelanggan dapat berpindah merek dengan mudah, ketika hambatan untuk meninggalkan pasar tinggi, ketika biaya tetap tinggi, ketika produk mudah rusak, ketika perusahaan pesaing berbeda dalam hal strategi, tempat mereka berasal, dan budaya, serta ketika *merger* dan akuisisi menjadi umum dalam suatu industri. Ketika persaingan antar perusahaan sejenis semakin intensif, laba perusahaan menurun, dalam beberapa kasus bahkan membuat suatu industri menjadi sangat tidak menarik.

2. Kemungkinan masuknya pesaing baru

Ketika perusahaan baru dapat dengan mudah masuk ke dalam industri tertentu, intensitas persaingan antarperusahaan meningkat. Tetapi, hambatan untuk masuk, dapat mencakup kebutuhan untuk mendapatkan teknologi dan pengetahuan khusus, kurangnya pengalaman, tingginya kesetiaan pelanggan, kuatnya preferensi merek, besarnya kebutuhan akan modal, kurangnya jalur distribusi yang memadai, peraturan pemerintah, tarif, kurangnya akses terhadap bahan mentah, kepemilikan paten, lokasi yang kurang menguntungkan, serangan balasan dari perusahaan yang sudah mapan, dan potensi kejenuhan pasar.

Di samping berbagai hambatan masuk, perusahaan baru kadang-kadang memasuki suatu bisnis dengan produk berkualitas lebih tinggi, harga lebih rendah, dan sumber daya pemasaran yang besar. Dengan demikian, tugas penyusun strategi adalah untuk mengidentifikasi perusahaan yang berpotensi masuk ke pasar, untuk memonitor strategi pesaing baru, untuk membuat serangan balasan apabila dibutuhkan, serta untuk memanfaatkan kekuatan dan peluang yang ada saat ini.

3. Potensi pengembangan pasar

Dalam banyak industri, perusahaan bersaing dekat dengan produsen produk substitusi dalam industri yang berbeda. Keberadaan produk substitusi menciptakan batas harga tertinggi yang dapat dibebankan sebelum konsumen beralih ke produk substitusi.

Tekanan kompetisi yang berasal dari produk substitusi meningkat sejalan dengan menurunnya harga relatif dari produk substitusi dan sejalan dengan biaya konsumen untuk beralih ke produk lain menurun. Cara terbaik untuk mengukur kekuatan kompetitif produk substitusi adalah dengan memantau pangsa pasar yang didapat oleh produk-produk tersebut, juga dengan memantau rencana perusahaan untuk meningkatkan kapasitas dan penetrasi pasar.

4. Kekuatan tawar-menawar penjual/pemasok

Kekuatan tawar-menawar pemasok (*bargaining power of supplier*) mempengaruhi intensitas persaingan dalam suatu industri, khususnya ketika ada sejumlah besar pemasok, ketika hanya ada sedikit barang substitusi yang cukup bagus, atau ketika biaya untuk mengganti bahan baku sangat mahal. Sering kali kepentingan yang dicari oleh pemasok dan produsen adalah saling memberikan harga yang masuk akal, memperbaiki kualitas, mengembangkan jasa baru, pengiriman *just-in-time*, dan mengurangi biaya persediaan, dengan demikian memperbaiki profitabilitas jangka panjang untuk semua pihak.

Perusahaan dapat menjalankan strategi integrasi ke belakang (*backward integration*) untuk mendapatkan kendali atau kepemilikan dari pemasok. Strategi ini efektif khususnya ketika pemasok tidak dapat diandalkan, terlalu mahal, atau tidak mampu memenuhi kebutuhan perusahaan secara konsisten. Perusahaan umumnya dapat menegosiasikan syarat yang lebih menguntungkan bagi pemasok ketika integrasi kebelakang merupakan strategi yang digunakan secara umum diantara perusahaan-perusahaan yang bersaing dalam suatu industri.

5. Kekuatan tawar-menawar pembeli/konsumen

Ketika konsumen terkonsentrasi atau besar jumlahnya, atau membeli dalam jumlah besar, kekuatan tawar-menawar mereka menjadi kekuatan utama yang mempengaruhi intensitas persaingan dalam suatu industri. Perusahaan pesaing mungkin menawarkan garansi yang lebih panjang atau jasa khusus untuk mendapatkan kesetiaan pelanggan ketika kekuatan tawar-menawar konsumen (*Bargaining power of consumer*) cukup besar. Kekuatan tawar-menawar konsumen juga lebih tinggi ketika yang dibeli adalah produk standar atau tidak terdiferensiasi. Ketika kondisinya seperti ini, konsumen sering kali dapat bernegosiasi tentang harga jual, cakupan garansi, dan paket aksesori hingga ke tingkat yang lebih tinggi bagi perusahaan besar sekalipun.

2.5.2 Analisis SWOT (*Strength, Weakness, Opportunity, Threat*)

Menurut Kotler (2003, p102) analisis SWOT merupakan evaluasi terhadap keseluruhan kekuatan, kelemahan, peluang dan ancaman. Analisis ini dibagi ke dalam dua bagian yaitu analisis lingkungan eksternal (peluang dan ancaman) dan analisis lingkungan internal (kekuatan dan kelemahan).

Menurut Pearce dan Robinson (2000, p202-204), analisis SWOT adalah analisis yang berdasarkan pada anggapan bahwa suatu strategi yang efektif berasal dari sumber daya internal suatu perusahaan (*Strengths* dan *Weaknesses*), dan sumber daya eksternal suatu perusahaan (*Opportunities* dan *Threats*).

- *Strength* (Kekuatan)

Suatu keunggulan sumber daya yang relatif terhadap pesaing dan kebutuhan dari pasar yang dilayani atau hendak dilayani oleh perusahaan kekuasaan yang dimiliki oleh suatu perusahaan dibandingkan dengan pesaing.

- *Weakness* (Kelemahan)

Keterbatasan atau kekurangan dalam sumber daya, keterampilan dan kemampuan yang secara serius menghalangi kinerja efektif perusahaan. Keterbatasan dalam fasilitas, sumber daya keuangan, kemampuan manajemen, keterampilan pemasaran merupakan sumber dari kelemahan.

- *Opportunity* (Peluang)

Adalah suatu daerah kebutuhan pembeli dimana perusahaan dapat beroperasi secara menguntungkan dan untuk merebut lebih banyak konsumen dibandingkan dengan para pesaing.

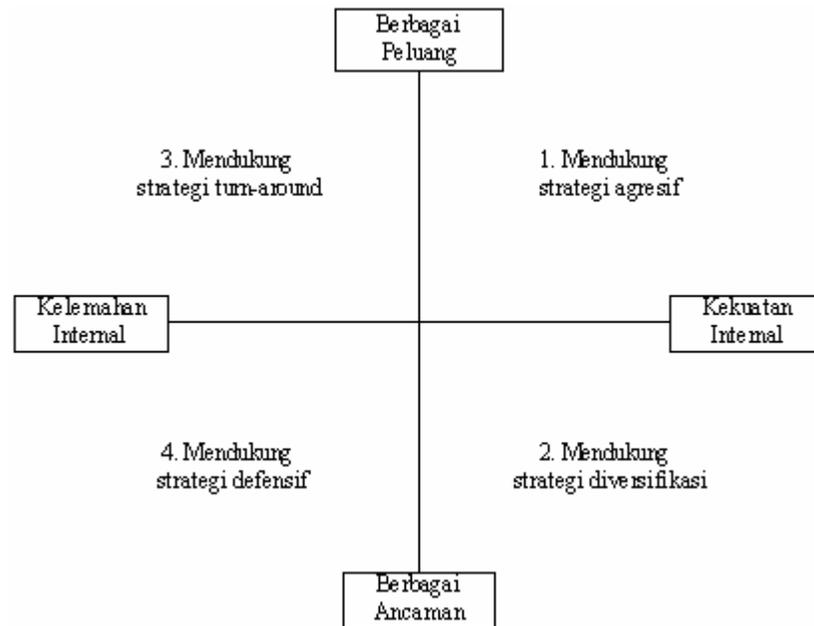
- *Threats* (Ancaman)

Tantangan dan ancaman yang dihadapi oleh suatu perusahaan dari para pesaing dalam merebut konsumen.

Analisis SWOT dapat digunakan dengan berbagai cara untuk meningkatkan analisis dalam usaha penetapan strategi. Umumnya yang sering digunakan adalah sebagai kerangka kerja/paduan sistematis dalam diskusi untuk membahas kondisi alternatif dasar yang mungkin menjadi pertimbangan perusahaan.

2.5.3 Diagram SWOT (*Strength, Weakness, Opportunity, Threat*)

Setelah didapat hasil tabel bobot skor dari masing-masing IFAS dan EFAS, langkah selanjutnya adalah memasukkan angka total bobot skor tersebut ke dalam diagram analisis SWOT berikut ini :



Gambar 2.8 Diagram SWOT

Keterangan :

Kuadran 1 : Ini merupakan situasi yang sangat menguntungkan, perusahaan tersebut memiliki peluang dan kekuatan sehingga dapat memanfaatkannya peluang yang ada, strategi yang harus diterapkan dalam kondisi ini adalah mendukung kebijakan pertumbuhan yang agresif (*Growth Oriented Strategy*)

Kuadran 2 : Meskipun menghadapi berbagai ancaman, perusahaan ini masih memiliki kekuatan segi internal, strategi yang harus diterapkan adalah menggunakan kekuatan untuk memanfaatkan peluang jangka panjang dengan cara strategi diversifikasi (produk/pasar)

Kuadran 3 : Perusahaan menghadapi peluang pasar yang sangat besar, tetapi di lain pihak ia menghadapi beberapa kendala/kelemahan internal. Fokus strategi perusahaan ini adalah meminimalkan masalah-masalah internal perusahaan sehingga dapat merebut peluang pasar yang lebih baik.

Kuadran 4 : Ini merupakan situasi yang sangat tidak menguntungkan, perusahaan tersebut menghadapi berbagai ancaman dan kelemahan internal.

2.5.4 Matriks SWOT (*Strength, Weakness, Opportunity, Threat*)

Menurut David (2004, p288-290), matriks *Strength-Weakness-Opportunity-Threat* (SWOT) merupakan alat pencocokan yang penting yang membantu manajer mengembangkan empat tipe strategi:

- Strategi SO atau Strategi Kekuatan-Peluang menggunakan kekuatan internal perusahaan untuk memanfaatkan peluang eksternal. Semua manajer menginginkan organisasi mereka berada dalam posisi di mana kekuatan internal dapat dipakai untuk memanfaatkan tren dan peristiwa eksternal.
- Strategi WO atau Strategi Kelemahan-Peluang bertujuan untuk memperbaiki kelemahan dengan memanfaatkan peluang eksternal. Kadang-kadang peluang eksternal yang besar ada, tetapi kelemahan internal sebuah perusahaan membuatnya tidak mampu memanfaatkan peluang itu.

- Strategi ST atau Strategi Kekuatan-Ancaman menggunakan kekuatan perusahaan untuk menghindari atau mengurangi dampak ancaman eksternal. Hal ini tidak berarti bahwa organisasi yang kuat pasti selalu menghadapi ancaman frontal dalam lingkungan eksternal.
- Strategi WT atau Strategi Kelemahan-Ancaman merupakan taktik defensif yang diarahkan untuk mengurangi kelemahan internal dan menghindari ancaman eksternal. Sebuah organisasi yang dihadapkan pada berbagai ancaman eksternal dan kelemahan internal, sesungguhnya dalam posisi yang berbahaya. Faktanya, perusahaan seperti itu mungkin harus berjuang agar dapat bertahan, atau melakukan merger, rasionalisasi, menyatakan pailit atau memilih dilikuidasi.

	Kekuatan (S)	Kelemahan (W)
Peluang (O)	<p><u>Strategi SO</u> Gunakan kekuatan untuk memanfaatkan peluang</p>	<p><u>Strategi WO</u> Atasi kelemahan dengan memanfaatkan peluang</p>
Ancaman (T)	<p><u>Strategi ST</u> Gunakan kekuatan untuk menghindari ancaman</p>	<p><u>Strategi WT</u> Meminimalkan kelemahan dan menghindari ancaman</p>

Gambar 2.9 Matriks SWOT
Sumber: David (2002, p186)