

BAB II

LANDASAN TEORI

2.1 Kategori Jasa

Menurut Buffa ; Sarin (2006), Terdapat empat kategori perusahaan jasa:

1. Jasa pribadi stagnan (Stagnant personal services)

Jasa ini menuntut kontak langsung antara pelanggan dan penyedia jasa. Tantangan memanajemeni jasa pribadi stagnan adalah meningkatkan efektifitasnya melalui manajemen yang lebih baik. Peningkatan produktifitas yang cukup bearti dalam kegiatan-kegiatan penunjang yang diperlukan untuk menyediakan jasa ini seringkali dapat terwujud.

2. Jasa pribadi yang dapat digantikan

Jasa ini menuntutu kontak pribadi lansung dan memiliki karakteristik yang serupa dengan jasa pribadi stagnan. Tetapi jasa jenis ini mungkin dapat digantikan dengan teknologi atau alternative lain.

3. Jasa progresif

Jasa ini mempunyai dua komponen. Satu komponen tidak banyak membutuhkan tenaga kerja dan pengurangan biaya yang cukup besar dapt dimungkinkan. Komponen kedua bersifat sangat padat karydan sangat mirip dengan jenis jasa pribadi stagnan. Jasa progresif dapat memperlihatkan pertumbuhan produktifitas dan reduksi biaya yang fenomenal di saat awal.

4. Jasa Eksplosif

Jasa ini tidak memerlukan kontak antara pelanggan dan tenaga kerja produksi.

Jasa jenis ini akan mengalami pertumbuhan produktivitas yang eksplosif sesuai dengan kemajuan teknologi.

2.2 Persediaan

Menurut Agus Sartono (2001) menerangkan bahwa persediaan yang ada dalam perusahaan akan tergantung pada jenis perusahaan yaitu;

1. Perusahaan Jasa

Persediaan perusahaan jasa biasanya timbul seperti persediaan bahan pembantu atau persediaan habis pakai.

2. Perusahaan manufaktur

Persediaan perusahaan manufaktur biasanya persediaan bahan pembantu, persediaan barang jadi, persediaan barang dalam proses dan persediaan bahan baku.

2.3 Fungsi Persediaan

Menurut Prawirosentono (2000), persediaan diadakan mulai dari bahan baku sampai barang jadi, antara lain berguna untuk:

- a. Mengurangi resiko keterlambatan datangnya bahan-bahan yang dibutuhkan untuk menunjang proses barang diperusahaan.
- b. Mengurangi resiko penerimaan bahan baku yang dipesan tetapi tidak sesuai dengan pesanan sehingga harus dikembalikan
- c. Menyimpan barang/bahan secara musiman sehingga dapat digunakan seandainya barang/bahan tersebut tidak tersedia di pasaran.

- d. Mempertahankan stabilitas operasi barang perusahaan yang berarti menjamin kelancaran proses barang
- e. Upaya penggunaan mesin yang optimal, karena terhindar dari terhentinya operasi barang karena tidak adanya persediaan (stock out)
- f. Memberikan pelayanan kepada konsumen dengan baik, di mana keinginan konsumen pada suatu waktu dapat dipenuhi dengan memberikan jaminan tetap tersedianya barang jadi tersebut.

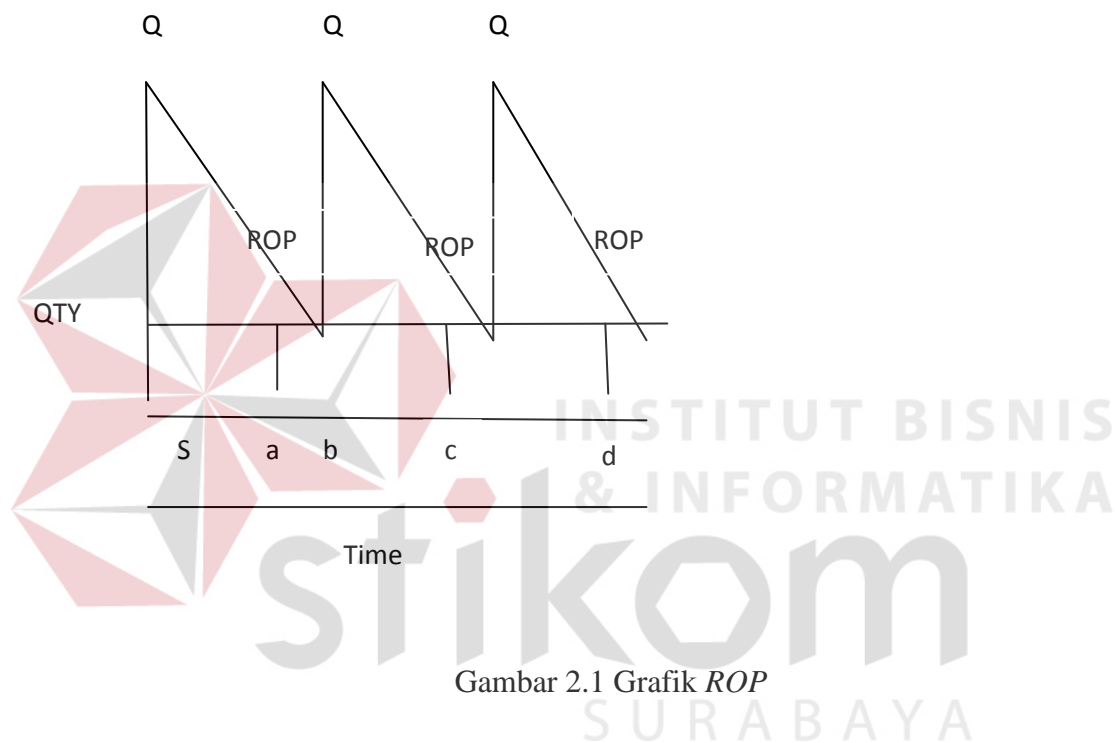
2.4 Pengelolaan

Menurut George R Terry dalam Ishak Arif (2009) menyatakan bahwa pengelolaan adalah kegiatan yang merencanakan, melaksanakan, mengontrol dan mengawasi, serta melakukan evaluasi unsur-unsur dasar manusia, benda-benda, mesin-mesin, metode-metode, uang dan pasar, memberikan kepemimpinan pada usaha-usaha manusia untuk mencapai tujuan dari badan usaha.

Pada tugas akhir ini, pengelolaan yang dibahas yaitu kegiatan merencanakan dan melaksanakan. Merencanakan didalam tugas akhir ini meliputi merencanakan *event* yang akan dibuat, barang yang akan digunakan, jadwal *event*, pelaksana *event*. Melaksanakan didalam tugas akhir ini meliputi melaksanakan jadwal *event* sesuai dengan waktunya, mengeluarkan barang yang dibutuhkan.

2.5 Titik Pesan Ulang (*Reorder Point/ ROP*)

Menurut Haryanto Tanuwijaya dan Henry Bambang Setyawan (2012), ROP adalah titik waktu dimana pemesanan dilakukan kembali, setelah persediaan mencapai jumlah tertentu, sehingga tidak terjadi kekurangan barang (bahan baku, komponen)



Gambar 2.1 Grafik ROP

Faktor utama yang perlu diperhatikan dalam menentukan *ROP* adalah kebutuhan rata rata per periode dan *lead time*, yaitu waktu antara dilakukannya pemesanan barang sampai dengan barang tersebut datang.

Bila D = permintaan kebutuhan barang per satuan waktu

L = *lead time*

Maka

$$ROP = D \times L \dots\dots\dots(3.2)$$

Apabila sistem persediaan membutuhkan stock pengaman dalam operasionalnya, maka rumus tersebut menjadi:

$$ROP = (D \times L) + SS \dots \dots \dots (3.3)$$

di mana SS menyatakan *safety stok* yang harus tersedia dalam sistem persediaan

2.6 Waktu Tunggu

Menurut Gaspersz (2006), dalam pemenuhan/pengisian kembali persediaan terdapat suatu perbedaan waktu yang cukup lama antara saat mengadakan pesanan (order) untuk pengisian kembali persediaan dengan saat penerimaan barang-barang yang dipesan tersebut diterima dan dimasukkan ke dalam persediaan (stock), perbedaan inilah yang dinamakan lead time.

2.7 Sistem

Sistem merupakan suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain (Fatta, 2007 : 3).

Sistem menurut Fatta (2007 : 5 - 6) memiliki karakteristik yang dapat membedakan suatu sistem dengan sistem lainnya, yaitu:

1. Batasan (*boundary*), penggambaran suatu elemen yang termasuk di dalam dan di luar sistem.
2. Lingkungan (*environment*), segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan *input* terhadap suatu sistem.
3. Masukan (*input*), sumber daya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.

4. Keluaran (*output*), sumber daya atau produk (informasi, laporan, dokumen, tampilan *layer computer*, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*), kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan *input* menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.
6. Penghubung (*interface*), tempat di mana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*), suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

Pendekatan sistem merupakan penerapan dari sistem ilmiah dalam manajemen, melalui cara ini akan diketahui faktor-faktor yang memengaruhi perilaku dan keberhasilan suatu organisasi atau sistem. Pendekatan sistem sendiri dapat ditandai dalam semua faktor penting yang ada untuk mendapatkan penyelesaian masalah dengan solusi terbaik dan dibuat suatu model kuantitatif untuk membantu keputusan secara rasional (Marimin dkk., 2006 : 3).

Suatu pendekatan sistem memiliki delapan unsur, meliputi : metodologi untuk perencanaan dan pengelolaan, suatu tim yang multi-disipliner, pengorganisasian, disiplin untuk bidang yang non-kuantitatif, teknik model matematik, teknik simulasi, teknik optimasi, dan aplikasi komputer (Marimin dkk., 2006 : 4).

2.8 Aplikasi

Aplikasi merupakan suatu kelompok *file* (*form*, *class*, *report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya Aplikasi *Payroll*, Aplikasi *Fixed Asset*. Ruang lingkup dari aplikasi berbeda-beda dari satu perusahaan ke perusahaan lainnya (Santoso, 2005 : 9).

Menurut Santoso (2005 : 10) aplikasi berdasarkan jenisnya digolongkan menjadi:

1. *Single Aplikasi - Single Project*, satu jenis aplikasi yang dikelompokkan ke dalam satu *project*. Secara umum *file* yang terbentuk akan diletakkan dalam satu *folder* dan harus dipastikan *file* tidak tersimpan pada *default folder*.
2. *Multi Aplikasi - Single Project*, suatu aplikasi tambahan dengan tugas yang berbeda disertakan pada suatu *project*. Secara umum *file* digabung pada satu *folder* atau *file* dari aplikasi dapat tercampur dengan *file* dari aplikasi lain.
3. *Multi Aplikasi - Multi Project*, setiap aplikasi dikelompokkan pada *project* tertentu dan *file* aplikasi disimpan pada *folder* tertentu pula, sehingga aplikasi yang terbentuk sebenarnya individu aplikasi yang berdiri sendiri dan dapat dikompilasi menjadi individu *file.exe*.
4. *Multi Aplikasi - Multi Project* dalam satu logika program, di mana *multi* aplikasi dan *multi project* digabungkan dalam satu grup. Dalam hal ini harus ditambahkan satu *project* baru untuk memanggil aplikasi lainnya.







2.9 Bagan Alir Dokumen

Definisi bagan alir dokumen menurut Jogiyanto (2001:795) adalah bagan yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan untuk mengkomunikasikan aliran data dan prosedur







proses informasi yang diperlukan dalam sistem informasi. Penggunaannya dilakukan dengan menggunakan simbol-simbol yang dihubungkan dengan panah-panah untuk menunjukkan aktivitas proses informasi.

Terdapat dua jenis di dalam bagan alir dokumen, yaitu *System Flow* dan *Document Flow*. *Document Flow* yaitu bagan alir formulir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. *System Flow* adalah bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem.

Tabel 2.1 Simbol *Document Flow*

Simbol	Nama	Keterangan
	Terminator	Digunakan untuk memulai atau mengakhiri program
	Decision	Perbandingan pernyataan untuk melanjutkan langkah berikutnya
	Dokumen	Sumber data
	Manual proses	Digunakan untuk mendefinisikan proses yang dilakukan secara manual
	Aliran dokumen	Digunakan untuk mengetahui aliran dokumen di dalam suatu program
	Penyimpanan file	Digunakan untuk mendefinisikan dokumen yang diarsipkan

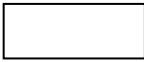
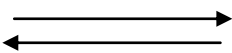
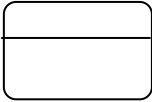

Tabel 2.2. Simbol *System Flow*

Simbol	Nama	Keterangan
	Input data	Digunakan untuk melakukan proses input data
	Decision	Perbandingan pernyataan untuk melanjutkan langkah berikutnya
	Dokumen	Sumber data
	Proses	Digunakan untuk mendefinisikan proses yang dilakukan oleh sistem
	Aliran data	Digunakan untuk mengetahui aliran data di dalam suatu program
	Penyimpanan data	Digunakan untuk mendefinisikan data-data yang disimpan ke dalam data base

2.10 Diagram Aliran Data

Diagram aliran data adalah model yang dibuat agar penganalis sistem dapat merepresentasikan proses-proses data di dalam organisasi. Pendekatan aliran data menekankan logika yang mendasari sistem. Dengan menggunakan simbol-simbol diagram aliran data, penganalis sistem dapat menciptakan suatu gambaran proses-proses yang bisa menampilkan dokumentasi yang solid.

Tabel 2.3. Simbol *Data Flow Diagram*

Simbol	Nama	Keterangan
	Entitas eksternal	Unit yang berinteraksi dengan sistem
	Aliran data	
	Proses	Proses untuk melakukan transformasi data
	Data store	Tempat penyimpanan data

2.11 Konsep Dasar Basis Data

Database Menurut Yuswanto (2005 : 2) *database* merupakan sekumpulan data yang berisi informasi yang saling berhubungan. Pengertian ini sangat berbeda antara *database* Relasional dan Non Relasional. Pada *database* Non Relasional, sebuah *database* hanya merupakan sebuah *file*.

Menurut Marlinda (2004 : 1) *database* adalah suatu susunan/ kumpulan data operasional lengkap dari suatu organisasi/ perusahaan yang diorganisir/ dikelola dan disimpan secara terintegrasi dengan menggunakan metode tertentu menggunakan komputer sehingga mampu menyediakan informasi optimal yang diperlukan pemakainya.

Penyusunan satu *database* digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu redundansi dan inkonsistensi data, kesulitan pengaksesan data, isolasi data untuk standarisasi, *multiple user* (banyak pemakai),

masalah keamanan (*security*), masalah integrasi (kesatuan), dan masalah data *independence* (kebebasan data).

2.12 Sistem Basis Data

Menurut Marlinda (2004:1), sistem basis data adalah suatu sistem menyusun dan mengelola record-record menggunakan komputer untuk menyimpan atau merekam serta memelihara dan operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi optimal yang diperlukan pemakai untuk proses mengambil keputusan.

Pada sebuah sistem basis data terdapat komponen-komponen utama yaitu Perangkat Keras (*Hardware*), Sistem Operasi (*Operating System*), Basis Data (*Database*), Sistem (Aplikasi atau Perangkat Lunak) Pengelola Basis Data (DBMS), Pemakai (*User*), dan Aplikasi (Perangkat Lunak) lain (bersifat opsional).

2.13 Database Management System (DBMS)

Menurut Kadir (2003 : 17) Definisi *Database Management System* (DBMS) pada sejumlah literatur sangatlah bervariasi. Secara umum, DBMS diartikan sebagai suatu program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memanipulasi, dan memperoleh data/ informasi dengan praktis dan efisien.

Dibandingkan dengan sistem yang berbasis kertas, DBMS memiliki empat unggulan:

1. Kepraktisan

Sistem yang berbasis kertas akan menggunakan kertas yang sangat banyak untuk menyimpan informasi, sedangkan DBMS menggunakan media penyimpan sekunder yang berukuran kecil tetapi padat informasi.

2. Kecepatan

Mesin dapat mengambil atau mengubah data jauh lebih cepat daripada manusia.

3. Mengurangi kejemuian

Orang cenderung menjadi bosan kalau melakukan tindakan-tindakan berulang yang menggunakan tangan (misalnya harus mengganti suatu informasi).

4. Kekinian

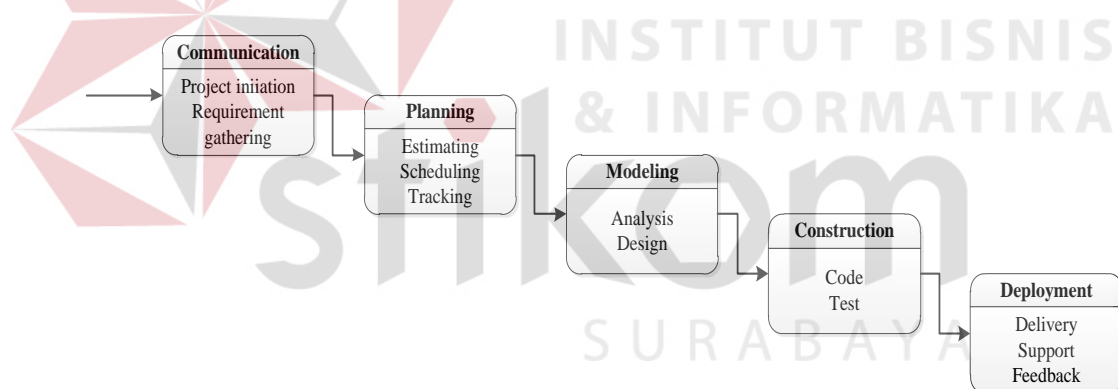
Informasi yang tersedia pada DBMS akan bersifat mutakhir dan akurat setiap saat.

Kebanyakan DBMS menyediakan mekanisme pengaturan sekuritas terhadap basis data berdasarkan wewenang pengguna. Sebagai contoh, si A hanya boleh membaca suatu data, yaitu untuk menghindari pengaksesan data sensitif oleh orang yang tidak berhak.

Di dalam lingkungan basis data, data lebih mudah digunakan. Pada beberapa DBMS tersedia fasilitas *query* yang memudahkan pengguna untuk memperoleh informasi. Pengguna yang tidak memiliki kemampuan pemrograman pun dengan mudah bisa menggunakan fasilitas *query* tersebut. Bagi pemrogram aplikasi, pembuat program aplikasi juga dapat dilakukan dengan mudah, jauh lebih mudah dibandingkan dengan kalau menggunakan bahasa-bahasa konvensional seperti *Cobol* dan *Fortran*. Alhasil, keadaan ini dapat meningkatkan produktifitas pemrogram.

2.14 System Development Life Cycle

Menurut Pressman (2015), System Development Life Cycle (SDLC) ini biasanya disebut juga dengan model waterfall. Menurut Pressman (2015), nama lain dari Model Waterfall adalah model air terjun kadang dinamakan siklus hidup klasik (classic life cycle), dimana hal ini menyiratkan pendekatan yang sistematis dan berurutan pada pengembangan perangkat lunak. Pengembangan perangkat lunak dimulai dari spesifikasi kebutuhan pengguna dan berlanjut melalui tahapan-tahapan perencanaan (planning), pemodelan (modeling), konstruksi (construction), serta penyerahan sistem perangkat lunak ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan berkelanjutan pada perangkat lunak yang dihasilkan.



Gambar 2.3 Pengembangan Menggunakan Model Waterfall (Pressman, 2015)

Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam Model Waterfall menurut Pressman (2015) :

a. *Communication*

Langkah pertama diawali dengan komunikasi kepada konsumen. Langkah awal ini merupakan langkah penting karena menyangkut penumpukan informasi tentang apa kebutuhan konsumen.

b. *Planning*

Setelah proses communication kita menetapkan rencana untuk pengerjaan software yang meliputi tugas-tugas teknis yang akan dilakukan, resiko yang mungkin terjadi, sumber-sumber yang dibutuhkan, hasil yang akan dibuat, dan jadwal pengerjaan.

c. *Modelling*

Pada proses modeling ini menerjemahkan syarat kebutuhan ke sebuah perancangan software yang dapat diperkirakan sebelum dibuat coding. Proses ini berfokus pada rancangan struktur data, arsitektur software, representasi interface, dan detail (algoritma) prosedural. Tahapan ini menghasilkan dokumen yang disebut software requirement.

d. *Construction*

Construction merupakan proses membuat kode. Coding atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Programmer akan menerjemahkan transaksi yang diminta oleh user. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu software, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

e. *Deployment*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah software atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan user. Kemudian software yang telah dibuat harus dilakukan pemeliharaan secara berkala.

2.15 Testing

Menurut Khan (2012:146), testing adalah proses untuk memeriksa atau mengevaluasi sistem atau komponen sistem secara manual atau terotomatisasi yang bertujuan untuk melakukan verifikasi bahwa sistem tersebut memenuhi persyaratan tertentu atau untuk mengidentifikasi perbedaan antara expected result dan actual result.

Menurut Lewis (2009:3), software testing adalah aktivitas menjalankan serangkaian eksekusi yang dinamis pada program software setelah source code software tersebut telah dikembangkan. Software testing dilakukan untuk menemukan dan memperbaiki sebanyak mungkin potensi kesalahan sebelum software tersebut digunakan oleh pelanggan atau end user.

Dari definisi di atas, testing merupakan aktivitas atau proses memeriksa dan mengevaluasi sistem dengan tujuan untuk menemukan kesalahan pada sistem tersebut.

2.16 *Black Box Testing*

Menurut Perry (2006:69), functional testing juga dapat disebut sebagai black box testing karena tidak ada pengetahuan dari logika internal sistem yang digunakan untuk membuat test case. Biasanya dalam pengujian fungsional, teknik validasi lebih digunakan untuk melakukan pengujian. Tim penguji melakukan validasi terhadap function key yang ada dan mengobservasi hasilnya.

Kelebihan dari functional testing:

1. Melakukan simulasi terhadap kegunaan sistem yang sebenarnya.
2. Tidak membuat asumsi terhadap struktur sistem.

Kekurangan dari functional testing:

1. Dapat berpotensi menghilangkan logika yang salah dalam software.
2. Memungkinkan terjadinya pengujian yang redundan.

Kedua metode ini berguna untuk validasi pada seluruh sistem. Seperti contohnya, sebuah functional test case dapat diambil dari dokumentasi tentang deskripsi bagaimana melakukan sebuah fungsi seperti menerima inputan bar code. Sebuah structural test case dapat diambil dari dokumentasi teknikal manual. Untuk pengujian sistem yang efektif diperlukan kedua metode tersebut.

Dari definisi di atas, black box testing adalah pengujian terhadap apa yang dilakukan oleh sistem, khususnya perilaku dan juga masalah bisnis di mana penguji membutuhkan informasi mengenai data input dan output yang diamati tetapi tidak perlu mengetahui struktur internal dari sistem tersebut.