

BAB III

ANALISIS TUGAS

3.1. ANALISA TUGAS

- a. Analisis tugas (Task Analysis) = suatu metode untuk menganalisis pekerjaan manusia, apa yang dikerjakan, dengan apa mereka bekerja, dan apa yang harus mereka ketahui.
- b. Contoh : apa saja tugas yang harus dilakukan dalam membersihkan rumah.
- c. Proses untuk menganalisis pekerjaan cara manusia, melakukan pekerjaannya : hal-hal yang mereka kerjakan, hal-hal yang mereka kenai tindakan, dan hal-hal yang harus mereka ketahui.
- d. Keluaran dari analisis tugas ini berupa perincian dari tugas yang dilakukan manusia.
- e. Hal-hal yang mereka gunakan, rencanakan, dan urutan tindakan yang biasa dilakukan untuk menyelesaikan tugas tergantung pada teknik yang digunakan.

3.1.1 Alasan Perlunya Analisis Tugas

Untuk memasukkan elemen manusia secara langsung pada perancangan secara sistematis dan terbuka sehingga dapat diperiksa secara teliti.

3.1.2 Istilah-Istilah Dalam Analisis Tugas

- a. Sasaran (*external task*) adalah kondisi sistem yang ingin dicapai manusia.
- b. Tugas (*Internal Task*) adalah himpunan terstruktur dari aktivitas yang dibutuhkan, digunakan atau dipercaya sebagai hal penting untuk mencapai sasaran dengan menggunakan perangkat tertentu.
- c. Aksi (*action*) adalah tugas yang tidak mengandung pemecahan persoalan atau komponen struktur terkendali.
- d. Rencana (*method*) terdiri atas sejumlah tugas atau aksi yang disusun dalam suatu urutan.

3.1.3 Maksud dari Analisa Tugas

Analisis tugas digunakan untuk :

- a. Manual dan pengajaran
 - 1. mengajarkan cara melakukan task
 - 2. menyusun manual atau materi ajar
 - 3. membantu user menjelaskan sistem ke orang lain
- b. Menangkap kebutuhan dan merancang sistem
 - 1. memandu perancangan
 - 2. membantu perancangan dalam memilih model internal
 - 3. meramalkan penggunaan sistem baru
- c. Merancang antarmuka detail
 - 1. mengklasifikasi tugas atau objek
 - 2. menghubungkan antara objek dengan aksi (OOP)

Contoh analisis tugas dalam rangka membersihkan rumah :

- a. Ambil penghisap debu
- b. Tancapkan penghisap debu pada alat penghubung listrik
- c. Bersihkan ruangan
- d. Jika kotak debu telah penuh, kosongkan
- e. Pasang kembali penghisap debu dan segala peralatan pembantunya.

Dalam kegiatan diatas, yang sebagai pengguna atau yang menjalankan kegiatan, harus mengetahui mengenai :

- a. Penghisap debu
- b. Alat-alat pembantu
- c. Kotak debu
- d. Lemari
- e. Ruangan, dan lain sebagainya.

3.2. TEKNIK ANALISA TUGAS

Perbedaan antara teknik analisis tugas dan teknik yang lain adalah bahwa teknik analisis tugas memiliki ruang lingkup yang luas. Selain meliputi tugas-tugas yang melibatkan penggunaan komputer, analisis tugas juga memodelkan aspek-aspek dunia nyata baik yang menjadi bagian maupun tidak menjadi bagian sistem komputer.

Teknik analisis tugas dibagi menjadi tiga bagian :

- a. Dekomposisi tugas
- b. Analisis berbasis pengetahuan
- c. Teknik berbasis relasi entitas.

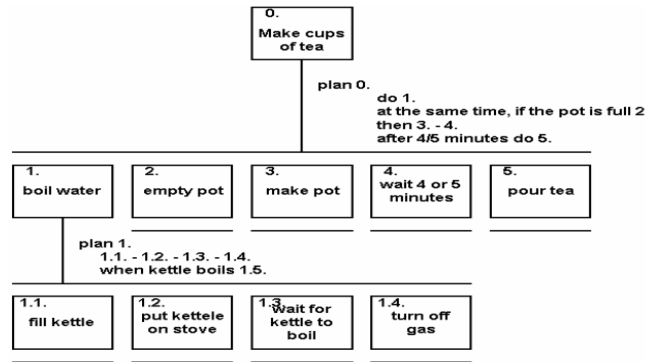
Analisa tugas berkaitan dengan sistem dan prosedur yang telah ada, dan alat utama yang digunakan adalah observasi dalam berbagai format. Salah satu tujuan analisis tugas adalah membantu pembuatan materi pelatihan dan dokumentasi lainnya. Pada saat dibutuhkan sistem baru, analisis tugas memberikan kontribusi pada proses identifikasi kebutuhan sistem. Dalam hal ini, analisis tugas memperjelas dan mengorganisasikan pengetahuan mengenai keadaan saat ini.

Dikaitkan dengan proses perancangan, analisis tugas termasuk dalam tahap awal pengidentifikasian kebutuhan, sedangkan model kognitif umumnya digunakan pada saat-saat akhir selama evaluasi.

3.2.1 Dekomposisi Tugas

- a. memilah tugas ke sub-tugas beserta urutan pelaksanaannya
- b. Salah satu pendekatan yang sering digunakan adalah *hierarchical task analysis* (HTA). Output HTA adalah hirarki tugas dan sub-task dan juga *plans* (rencana) yang menggambarkan urutan dan kondisi (syarat) suatu sub-tugas dilaksanakan.

Contoh : Membuat Air Teh



Hirarki tugas untuk membuat secangkir teh

Contoh pengelompokkan tugas / jenis perencanaan :

- fixed sequence* = contoh : 1.1-1.2-1.3
- Optional Tasks* = pada plan 0 'empty pot' dan pada plan 5.3. 'add sugar' mungkin tidak dilaksanakan tergantung dari situasinya.
- Waiting Events* = pada plan 1, harus menunggu ketel sampai mendidih, dan plan 0 menunggu 4 atau 5 menit
- Cycles* = kerjakan 5.1 – 5.2 jika cangkir masih kosong
- Time-sharing* = tugas 1 dan 2 dapat dilaksanakan dalam waktu yang bersamaan
- Discretionary* = kerjakan sembarang 3.1, 3.2, atau 3.3 dalam urutan sembarang
- Mixtures* = kebanyakan suatu rencana meliputi hal-hal diatas.

3.2.2 Analisa Berbasis Pengetahuan

Dimulai dengan mendaftar semua objek dan aksi yang terlibat dalam tugas dan kemudian membangun taksonominya, mirip seperti apa yang dilakukan pada bidang biologi: hewan termasuk dalam invertebrata dan vertebrata, hewan vertebrata adalah ikan, burung, reptil, amphibi, atau mamalia, dan seterusnya Tujuannya untuk memahami knowledge yang dibutuhkan untuk melaksanakan tugas.

3.2.3 Teknik Berbasis Relasi Entitas

Teknik berbasis relasi entitas biasanya berasosiasi dengan basis data pada model database entitas, mewakili sistem contoh tabel dan atribut pada analisis tugas, menekankan pada objek, aksi dan hubungannya diantaranya, mirip dengan analisis berbasis objek tetapi mengikutsertakan entitas non-komputer dan penekanan pada pemahaman domain, bukan implementasi.

3.3. SUMBER INFORMASI & PENGUMPULAN DATA

Analisis tugas yang akan dilakukan akan berhasil atau menghasilkan hasil yang baik jika didukung oleh sumber data yang cukup. Proses analisis data tidak hanya semata-mata mengumpulkan dan mengorganisasikan data dan mepresentasikan data, namun kadangkala kita harus melihat kembali sumber data tersebut dengan pertanyaan dan pandangan baru. Pada prakteknya, keterbatasan waktu dan biaya membuat analis berusaha mengumpulkan data secepat dan seekonomis mungkin. Bahkan jika mungkin seorang analis memanfaatkan semaksimal mungkin penggunaan sumber informasi yang murah yang sudah ada sebelum memanfaatkan sumber yang memakan biaya.

Berikut beberapa sumber informasi yang dpt diperoleh untuk membuat analisa tugas.

a. Dokumentasi yang tersedia

1. Lebih sering memuat penjelasan bagaimana tugas akan dikerjakan (dari pada bagaimana mereka akan mengerjakan tugas sekarang ini).
2. Standar
3. Manual
4. Historis
5. Praktek yang baik

b. Observasi

1. Di dalam observasi, lihat apa yang dikerjakan user
2. Merekam dengan videotape (untuk melihat kembali atau mengulangi)
3. Gunakan catatan, bagan
4. Fokus pada tugas khusus – lingkungan yang berhubungan dicatat, tetapi kemudian diubah menjadi sub-tugas abstrak

c. Wawancara

Bertanya pada seorang yang ahli pada bidang tugas yang akan dianalisis seringkali merupakan cara langsung yang cepat untuk mendapatkan informasi mengenai suatu tugas. Ahli tersebut bisa saja si manager, supervisor, atau staf yang memang mengerjakan tugas tersebut. Wawancara kepada ahli sebaiknya dilakukan setelah observasi. Hasil observasi dapat direfleksikan dengan wawancara untuk mengetahui perilaku atau kondisi yang diinginkan dan tidak diinginkan.

3.4 KOMPONEN DATA INPUT DAN OUTPUT

3.4.1 Data I / O

Dalam Analisis tugas kita harus mempunyai dua macam cara/system untuk melakukan analisa,yaitu :

- a. **INPUT** → Merupakan aktifitas pemberian data kepada komputer, dimana data tersebut merupakan masukan bagi komputer.
- b. **OUTPUT** → Keluaran, hasil dari suatu proses, baik berupa data maupun berbentuk informasi yang telah diolah.

3.4.2 Unit Input(*Input Device*)

Input Device atau alat masukan adalah perangkat/ media untuk memasukkan data dari luar ke dalam suatu memori dan processor untuk diolah guna menghasilkan informasi yang diperlukan. Input devices atau unit masukan yang umumnya digunakan personal computer (PC) adalah keyboard dan mouse, keyboard dan mouse adalah unit yang menghubungkan user(pengguna) dengan computer. Data yang dimasukkan ke dalam system computer dapat berbentuk signal input dan maintenance input. Sign input berbentuk data yang dimasukkan ke dalam system computer, sedangkan maintenance input terbentuk program yang digunakan untuk mengolah datayang dimasukkan. Jadi input device selain digunakan untuk memasukkan data dapat pula digunakan untuk memasukkan program.

3.4.3 Unit Output(*output device*)

Perangkat output atau *output devices* adalah alat yang digunakan untuk menampilkan informasi dari computer. Peralatan yang sering digunakan adalah monitor, printer, dan speaker.

3.5 REPRESENTASI DATA :

3.5.1 Pengertian :

Proses perubahan konsep-konsep yang abstrak maupun nyata dalam bentuk yg kongkret. Cara melakukan representasi data :

- a. Daftar, ringkasan, matriks
- b. Gunakan alat bantu bagan

- c. Tambahkan detail yang semakin bertambah
- d. Ketahui lebih lanjut berapa detail yang cukup
- e. Akankan ditambah ringkasan yang dihubungkan dengan sub-tugas khusus
- f. Baik untuk tugas yang terurut
- g. Tidak mendukung dengan baik tugas-tugas yang parallel
- h. Tidak mendukung dengan baik percabangan

3.5.2 Naratif:

- a. Menjelaskan tugas-tugas dalam bentuk kalimat
- b. Seringkali versi diperluas dari daftar atau ringkasan
- c. Lebih efektif untuk mengkomunikasikan ide-ide umum dari tugas
- d. Tidak efektif untuk detail
- e. Tidak efektif untuk tugas yang bercabang
- f. Tidak efektif untuk tugas parallel

3.5.3 Hierarki

Hierarki Task Analysis (HTA)

- a. Notasi grafik dan dekomposisi dari tugas
- b. Tugas merupakan kumpulan dari aksi
- c. Tugas diatur ke dalam rencana
- d. Mengelompokkan sub-tugas dengan berurut lebih disukai dan kondisi-kondisi prasyarat

Soal dan Pembahasan

Soal :

Sebutkan teknik dari analisa tugas

Pembahasan :

- a. Dekomposisi tugas
- b. Analisis berbasis pengetahuan
- c. Teknik berbasis relasi entitas.

BAB IV

DESAIN

4.1 TUJUAN DARI DESAIN

Suatu teknik yang andal untuk rancangan yang menghasilkan suatu sistem yang dapat dipakai.

Bagi perancang antarmuka, hal yang sangat penting untuk ia perhatikan adalah bahwa ia sebaiknya (atau seharusnya) mendokumentasikan semua pekerjaan yang ia lakukan. Dengan dokumentasi yang baik, ia dapat mengubah rancangannya ketika ia berubah pikiran atau menemukan bahwa rancangannya tidak mudah diimplementasikan. Perubahan yang terjadi bisa karena usulan pengguna yang berubah atau karena alasan lain.

Dokumentasi rancangan dapat dikerjakan atau dilakukan dengan beberapa cara :

- a. Membuat sketsa pada kertas
- b. Menggunakan peranti purwarupa GUI
- c. Menuliskan keterangan yang menjelaskan tentang kaitan antara satu jendela dengan jendela yang lain.
- d. Menggunakan peranti bantu yang disebut CAE.

Cara kedua dan cara keempat ini jarang untuk digunakan karena mahal, jadi cara ini digunakan untuk pembuatan antarmuka grafis untuk suatu pekerjaan dengan skala besar.

4.2 CARA PENDEKATAN

Program aplikasi pada dasarnya dapat dikelompokkan dalam 2 kategori besar yakni program aplikasi untuk keperluan khusus dengan pengguna yang khusus pula (special purpose software) dan program aplikasi yang akan digunakan oleh pengguna umum (general purpose software / public software)

Pada kelompok pertama, yakni pada program aplikasi untuk keperluan khusus, misalnya program aplikasi untuk keperluan khusus, misalnya program aplikasi untuk inventori gudang, pengelolaan data akademis mahasiswa, pelayanan reservasi hotel, dan program-program aplikasi yang serupa, kelompok calon pengguna yang akan memanfaatkan program aplikasi tersebut

dapat dengan mudah diperkirakan, baik dalam hal keahlian pengguna maupun ragam antarmuka yg akan digunakan.

Dalam kelp ini ada satu pendekatan yakni dengan pendekatan perancangan berpusat-ke-pengguna (*User-Centered Design Approach*).

Perancangan berpusat-ke-pengguna (*User-Centered Design Approach*). Berbeda dengan *user design approach*.

UCDA ini melibatkan pengguna, tetapi bukan dalam hal si pengguna memikirkan bagaimana implementasinya nanti, tetapi pengguna diajak aktif untuk berpendapat ketika perancang antar muka sedang menggambar “wajah” antarmuka yang mereka inginkan. Jadi disini si pengguna dan perancang antarmuka duduk bersama-sama untuk merancang wajah antarmuka yang diinginkan pengguna. Pengguna menyampaikan keinginannya sementara perancang menggambar keinginan pengguna tersebut sambil menjelaskan keuntungan dan kerugian wajah antarmuka yang diinginkan oleh pengguna. Dengan cara ini si pengguna sudah mendapat gambaran nyata mengenai antarmuka yang nanti akan mereka gunakan.

Pada pendekatan *user design approach*, pengguna sendirilah yang merancang wajah antarmuka yang diinginkan. Di satu sisi, cara ini akan mempercepat proses pengimplemntasian modul antarmuka. Tetapi di sisi yang lain, hal ini justru sangat memberatkan pemrogram karena apa yang diinginkan pengguna belum tentu dapat diimplementasikan dengan mudah atau bahkan tidak dapat dikerjakan dengan menggunakan peranti yang ada.

Untuk public software, perlu menganggap bahwa program aplikasi tersebut akan digunakan oleh pengguna dengan berbagai tingkat kepandaian dan karakteristik yang sangat beragam. Disatu sisi keadaan ini dapat dikatakan “memaksa”, memaksa pengguna untuk menerima hasil rancangan, namun hal ini akan berdampak pada tidak banyaknya pengguna thdp aplikasi ini. Satu kunci penting dalam pembuatan model antarmuka untuk program-program aplikasi pada kelompok ini adalah *customization*. Dengan *customization*, pengguna dapat menggunakan program aplikasi dengan wajah antarmuka yang sesuai dengan selera masing-masing pengguna.

4.3 PRINSIP DESAIN

4.3.2 Model Pengguna

Model mental pengguna merupakan model konseptual yang dimiliki oleh pengguna ketika ia menggunakan sebuah sistem atau program aplikasi.

Model ini memungkinkan seorang pengguna untuk mengembangkan pemahaman mendasar tentang bagian yang dikerjakan oleh program.

Dengan pertolongan model itu pengguna dapat mengantisipasi pengaruh suatu tindakan yang ia lakukan dan dapat memilih strategi yang cocok untuk mengoperasikan prog tersebut.

Model pengguna dapat berupa suatu simulasi tentang keadaan yang sebenarnya dalam dunia nyata, sehingga ia tidak perlu mengembangkannya sendiri dari awal.

4.3.3 Bahasa Perintah (*Command Language*)

Setelah pengguna mengetahui dan memahami model yang ia inginkan, dia memerlukan peranti untuk memanipulasi model itu.

Idealnya komputer kita mempunyai bahasa perintah yang alami, sehingga model pengguna dengan cepat dapat dioperasionalkan.

4.3.4 Umpan Balik

Maksudnya diartikan sebagai kemampuan sebuah program yang membantu pengguna untuk mengoperasikan program itu sendiri. Umpan balik dapat berbentuk pesan penjelasan, pesan penerimaan perintah, indikasi adanya objek terpilih, dan penampilan karakter yang diketikkan lewat papan ketik. Beberapa umpan balik ini ditujukan untuk yang belum berpengalaman.

Umpan balik dpt digunakan untuk memberi keyakinan bahwa program telah menerima perintah pengguna dan dpat memahami perintah tersebut.

4.3.5 Penampil Informasi

Komponen ini digunakan untuk menunjukkan status informasi atau program ketika pengguna melakukan suatu tindakan.

Perancang harus menampilkan informasi seefektif mungkin sehingga mudah dipahami.

4.4 IDE DALAM MEMBUAT DESAIN

Sekarang kita akan melihat bagaimana merencanakan suatu proyek. Untuk keperluan ini dibutuhkan suatu ide,. Ada beberapa tipe ide guna memulai suatu proyek :

1. Memperbaiki atau membuat sesuatu lebih baik, meniru atau memperbaiki sesuatu yg telah dibuat. Dengan berbekal ide ini maka pembuatan proyek akan memerlukan lebih sedikit langkah dan lebih sedikit waktu.
2. Ide baru dimana pemrogram akan membuat sesuatu yang benar-benar baru. Untuk menggunakan ide ini tentu diperlukan waktu dan langkah yang tidak mudah.
3. Kebutuhan pasar. Ide juga bisa didapat dengan melihat kebutuhan pasar.

Setelah semua ide terchecklist, sebaiknya seorang pemrogram tidak langsung mewujudkannya, tetapi ada beberapa hal yang harus diperhatikan :

- a. Sudah berapa banyak persiapan check list yang telah dibuat??
- b. Berapa biaya yang dibutuhkan
- c. Berapa luas penggunaannya.
- d. Berapa lama proses pembuatannya.
- e. Apakah ide ini lebih baik dari yang sudah ada

Jika sudah diputuskan untuk membuat suatu proyek aplikasi, tugas berikutnya adalah membuat rincian persyaratan proyek yg memberikan deskripsi rincian hasil akhir.

Jika persyaratan sudah dipetakan secara detail, baik fungsi input maupun outputnya, maka perlu ditinjau kembali apakah masih perlu diubah dan diperbaiki. Bila ternyata sudah baik, maka tiba saatnya kita memilih bahasa pemrograman yang tepat untuk membuat aplikasi tersebut.

Bahasa pemrograman yg digunakan sangat menentukan hasil yang akan diperoleh.

4.4.1 Tantangan Dalam Membuat Design

Antara lain: Proses pembuatan. Metoda merancang, Produk yang dihasilkan (rancangan), Serta disiplin ilmu yang digunakan (disain).

Soal dan Pembahasan

Soal :

Jelaskan point-point dari prinsip desain.

Pembahasan :

a. Model Pengguna

Model mental pengguna merupakan model konseptual yang dimiliki oleh pengguna ketika ia menggunakan sebuah sistem atau program aplikasi.

Model ini memungkinkan seorang pengguna untuk mengembangkan pemahaman mendasar tentang bagian yang dikerjakan oleh program.

Dengan pertolongan model itu pengguna dapat mengantisipasi pengaruh suatu tindakan yang ia lakukan dan dapat memilih strategi yang cocok untuk mengoperasikan prog tersebut.

Model pengguna dapat berupa suatu simulasi tentang keadaan yang sebenarnya dalam dunia nyata, sehingga ia tidak perlu mengembangkannya sendiri dari awal.

b. Bahasa Perintah (*Command Language*)

Setelah pengguna mengetahui dan memahami model yang ia inginkan, dia memerlukan peranti untuk memanipulasi model itu.

Idealnya komputer kita mempunyai bahasa perintah yang alami, sehingga model pengguna dengan cepat dapat dioperasionalkan.

c. Umpan Balik

Maksudnya diartikan sebagai kemampuan sebuah program yang membantu pengguna untuk mengoperasikan program itu sendiri. Umpan balik dapat berbentuk pesan penjelasan, pesan penerimaan perintah, indikasi adanya objek terpilih, dan penampilan karakter yang diketikkan lewat papan ketik. Beberapa umpan balik ini ditujukan untuk yang belum berpengalaman.

d. Penampil Informasi

Komponen ini digunakan untuk menunjukkan status informasi atau program ketika pengguna melakukan suatu tindakan.

Perancang harus menampilkan informasi seefektif mungkin sehingga mudah dipahami.