

BAB 6 FASE ANALISIS

6.1. PENDAHULUAN

Tujuan dari fase analisis adalah mendefinisikan secara tepat apa yang dapat dilakukan sistem untuk user, dan bagaimana sistem tersebut menyesuaikan dengan lingkungan user.

Aktivitas Pertama : (kejadian penting) dari fase ini adalah untuk menghasilkan dokumen yang menjelaskan arti lingkungan sistem, disebut **Functional Specifications (FS) / Spesifikasi Fungsi**.

Aktivitas kedua : Setelah mengerjakan FS, anda kini memiliki pengetahuan yang lebih dibandingkan pada Fase Definisi, sehingga anda harus meninjau ulang rencana permulaan proyek dan perkiraan awal.

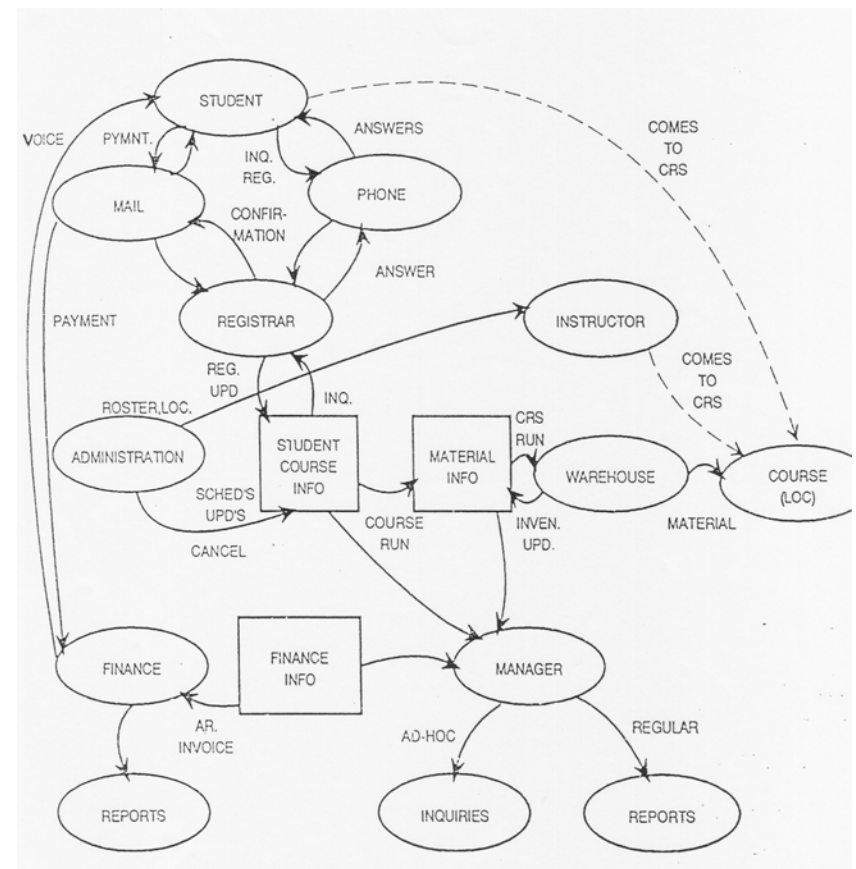
Aktivitas ketiga : menuliskan *development proposal* / proposal pengembangan, dan akan dikerjakan jika dua metode proposal akan dilakukan. Hal tersebut akan ditulis setelah FS.

Isi dan garis besar proposal pengembangan sama dengan proposal analisis, kecuali bahwa proposal pengembangan dikerjakan dengan menggunakan lima tahap dari pengembangan.

Dalam fase analisis “Anda harus menghadapi apa yang akan dilakukan, bukan mengenai bagaimana hal tersebut akan dilakukan, karena fase disain akan membahasnya”.

6.2. ALIRAN DATA YOURDON / METODE ANALISIS BUBBLE CHART (THE YOURDON DATA-FLOW/BUBBLE CHART METHOD OF ANALYSIS)

Edward Yourdon menemukan sebuah metode grafik untuk mendokumentasikan dan mengendalikan proses analisis yang menjadi sangat populer (*Referensi 11*). Gambar berikut ini adalah sebuah aplikasi dari metode tersebut untuk proyek ABC.



Pendefinisian User

Analisis bersama-sama dengan user mengembangkan diagram seperti pada gambar 6.1. Mereka mulai dengan membuat daftar semua user yang akan memiliki hubungan dengan sistem. Termasuk user tidak langsung seperti *STUDENT*.

Kemudian mereka menggambar garis panah untuk semua input dan output untuk masing-masing user, garis diberi nama dengan informasi atau data yang melewatinya.

Garis panah tersebut mewakili aliran informasi (*STUDENT* → *REGISTRAR* melalui telepon), aliran data (*REGISTRAR* → *COMPUTER* lewat terminal) atau kejadian perpindahan secara fisik dari bagian-bagian (*WAREHOUSE* → *CLASSROOM ships material*).

Inilah sebabnya mengapa diagram ini disebut diagram 'aliran data'. Kemudian analisis dan user mengidentifikasi informasi umum yang disimpan oleh sistem (informasi kursus, informasi murid, informasi material) dan menuliskannya ke dalam lingkaran.

Pendefinisian Antarmuka User

User dan Analisis menjelaskan setiap bagian yang diwakili oleh garis panah, yang merupakan aliran data antara user dan sistem. Hal ini akan mengontrol penjelasan mengenai semua menu, formulir, laporan, perintah-perintah dan pesan-pesan – dengan kata lain merupakan 'tampilan antarmuka user' pada sistem.

Tujuan dari proses ini adalah :

- Pertama untuk menjelaskan tampilan antarmuka pada komputer.
- Kedua untuk memperoleh pemahaman yang umum dari bisnis user. Seringkali user belajar mengenai bisnisnya sendiri dari tipe analisis ini.

Sebagai contoh, analisis dari aliran data *STUDENT* ke *REGISTRAR* akan dihasilkan sebagai berikut :

STUDENT → *REGISTRAR* and *REGISTRAR* → *STUDENT*

Method : Verbal over phone, or mailed in

Inquiries

Location, dates of courses

Number enrolled/maximums

Cost

.....

Responses

Course locations, dates (next 6 months)

Number enrolled (next 6 months); maximum allowed

Cost

.....

Changes

Update name, address, payment information of

student

Cancel a student from a course

Register a student

Obtain and enter name, address, course (by number)

Payment information

Performance

Must handle up to 3 calls per minute

Analisis terhadap REGISTRAR → ABC akan menghasilkan :

REGISTRAR → ABC

Method : Terminal input

Automatic registrar menu

When registrar logs in with specific account number, menu of

The format in the Functional Specification Figure 3.9. is presented.

To make a choice on this menu, the registrar can use either the UP

and DOWN arrows keys followed by RETURN, or move the mouse up

or down, followed by press on mouse button.

If student wishes information on course

Registrar chooses 1.

Menu of format FS Fig. 3.10 appears.

If student wishes to enroll.....

Langkah berikutnya adalah merinci seluruh menu, formulir, laporan dan perintah yang tepat. Semua menu seperti *REGISTRAR* dan pertanyaan mengenai kursus harus dijelaskan.

6.3. SPESIFIKASI FUNGSI (THE FUNCTIONAL SPECIFICATIONS / FS)

FS menjelaskan semua tingkah laku sistem dalam bentuk cerita dan gambar. Definisikan antarmuka user seperti di atas, menu-menu, perintah-perintah, respon, laporan dan pesan-pesan dijelaskan sebanyak mungkin. Setiap perubahan di dalam lingkungan user karena sistem baru akan dijelaskan.

Semua pengiriman, termasuk hardware, software, pelatihan, dokumentasi dan garansi dirinci.

Sebagai tambahan pada proposal, FS juga merupakan kontrak antara User dengan Tim Proyek (PT). Sejumlah uang yang besar mungkin dipertaruhkan, dan user membutuhkan lebih rinci tentang apa yang dapat diberikan dibandingkan apa yang ada di proposal. FS mungkin akan dinegosiasikan dan ditinjau kembali, dan ketika persetujuan dicapai proposal harus ditanda tangani oleh kedua belah pihak.

Garis Besar FS (Outline of the FS)

1. Judul Halaman (*Title Page*)

Judul fungsi spesifikasi, nama sistem, pembuat, dan tanggal Jangan lupa nomor versi : dokumen ini akan direvisi !

2. Daftar Isi (*Table of Contents*)

Nama bagian, berikut nomor halaman

3. Gambaran Sistem / Ikhtisar Sistem (*System Overview*)

Menjelaskan sistem yang akan dibuat. Ingatlah bahwa FS adalah **dokumen teknik yang ditujukan untuk pembaca non teknis (user)**. Cara terbaik untuk menjelaskannya dengan menggunakan gambar.

Marilah kita ambil contoh **sistem *Amalgamated Basketweaving Course (ABC)*** yang dijelaskan di awal. Sistem berdasarkan data mengenai kursus (*Course*) dan murid (*Student*).

User membutuhkan keterangan yang pasti mengenai data pendaftaran, kursus yang masih dibuka/tersedia, jadwal, rincian akuntansi, dsb.

User juga membutuhkan kemampuan untuk merubah data. User membutuhkan laporan yang dihasilkan, seperti faktur, konfirmasi, jumlah murid yang mendaftar.

Semua bagian ini harus ada antarmukanya dengan user, sehingga sebaiknya dibuatkan sistem menu menggunakan *mouse*.

Untuk menjelaskan semua ini, anda sebaiknya mulai dengan diagram seperti pada gambar 6.2.

Gambar. 6.2. Major functions of the system.

4. Tujuan Utama (*Major Objectives*)

Buatlah daftar tujuan sistem, hubungkan masing-masing ke modul utama. Contoh *INQUIRY* akan menjawab pertanyaan seperti “Berapa banyak murid yang mendaftar kursus”.

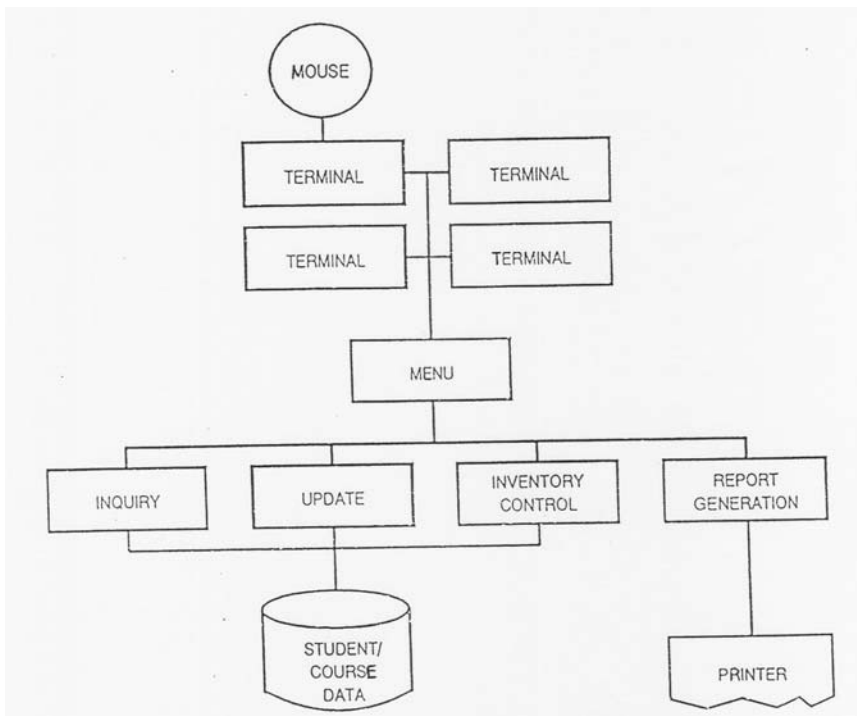
Menjelaskan bagaimana sistem yang baru akan mempengaruhi lingkungan user, yaitu dimana terminal akan ditempatkan, siapa yang menggunakannya, laporan apa yang akan dibuat, kapan dan bagaimana hal ini akan mengubah pekerjaan setiap orang. Anda harus memperingatkan user apabila sistem ini akan mempengaruhi berbagai aspek kehidupannya.

5. Kebutuhan Khusus Sistem (*Special System Requirements*)

Bagian ini menunjukkan kebutuhan-kebutuhan sistem seperti jaringan, kesesuaian, keamanan, ketahanan, dan kemudahan dalam menggunakan sistem.

Persoalan yang rumit seperti **respon** (jumlah waktu dalam detik yang dibutuhkan komputer untuk menjawab), **throughput** (jumlah total pekerjaan yang diselesaikan komputer dalam jangka waktu tertentu) dan **growth** / perkembangan (kebutuhan sistem untuk beberapa tahun ke depan) dapat ditunjukkan disini.

Sebagai contoh bagaimana jika RD berisi pertanyaan seperti : “Sistem harus memberikan respon untuk setiap input dalam 5 detik”. Sebuah komputer tercepat yang



pernah dibuat sekalipun membutuhkan waktu lebih dari 5 detik untuk merespon berbagai permintaan.

Demikian pula jangan menjanjikan dengan pasti mengenai *throughput* atau *growth*. Janji-janji yang pasti dapat diberikan pada sejumlah hal, seperti jumlah user, ukuran file, transaksi per-menit, atau pengembangan hardware, akan tetapi hal-hal tersebut mungkin sulit dipenuhi pada waktu penerimaan.

6. Deskripsi Komponen (*Component Descriptions*)

Bagian ini menjelaskan secara detail masing-masing isi kotak, atau fungsi yang terdapat pada gambar 6.2.

Jangan menjelaskan file yang berorientasi informasi seperti organisasi file, record dan field – semua itu sudah ada dalam disain. Lakukan pernyataan yang menunjukkan batasan, seperti jumlah maksimum kursus yang dapat ditangani oleh sistem.

7. Pengiriman yang lain (*Other Deliverables*) Dokumentasi.

Menyatakan jumlah dokumen yang dihasilkan, pembaca yang diharapkan, dan kegunaannya.

User's Guide sebaiknya menyediakan 2 tujuan :

- Pertama, sebagai alat pembelajaran
- Kedua, sebagai referensi dengan petunjuk seluruh perintah dan pesan yang akan disajikan secara alfabet.

Pelatihan. Buatlah daftar modul-modul atau topik-topik yang menjadi *cover* pada masing-masing kursus, dan materi pelatihan yang digunakan.

8. Perubahan Spesifikasi (*Specification Changes*)

Perubahan FS mungkin menyebabkan perubahan ke seluruh item-item yang lain, yang menyebabkan biaya menjadi mahal dan penundaan waktu pengiriman. Perubahan harus diminimalkan.

9. Penerimaan (*Acceptance*)

Salah satu masalah terbesar dalam dunia software adalah user kadang-kadang enggan untuk menerima dan membayar sistem tersebut. Oleh karena itu dalam FS kita rinci metode penerimaan, dan mengakhirinya dengan baik.

10. User dan Interface Tim Proyek (*User and Project Team Interface*)

User dan Tim proyek harus saling berkomunikasi pada level teknik maupun manajemen. Kebutuhan secara teknik dari User diperlukan saat Tim proyek memerlukan jawaban yang cepat dan akurat berbagai pertanyaan yang bersifat teknik. Berbagai pertanyaan ini tidak selesai hanya pada fase analisis, tetapi akan semakin kompleks saat proyek dilaksanakan. Sebaiknya user menunjuk paling sedikit satu orang yang dapat menjawab pertanyaan-pertanyaan tersebut.

User dan Tim proyek harus berkomunikasi pada level manajemen dengan baik. Hal ini harus dilakukan paling

tidak oleh Proyek Koordinator User dan Manajer proyek. Mereka akan mendiskusikan berbagai isu seperti pendanaan, jadwal, perubahan-perubahan, dan masalah-masalah sumber daya manusia.

11. Tanggung Jawab User (*User's Responsibilities*)

Untuk menghemat uang dan waktu, atau jika user berharap dilibatkan lebih banyak, Tim proyek mungkin meminta kepada user untuk mengerjakan tugas-tugas proyek, seperti menyediakan *data test*, menulis *User's Guide*, atau bahkan merencanakan *acceptance test*. Buatlah daftar seluruh kegiatan dan batas waktunya. Ingatkan user untuk menandatangani dokumen ini.

12. Istilah, Kondisi dan Asumsi (*Terms, Condition and Assumptions*)

Buatlah daftar aturan baru dan kebijaksanaan yang harus dipatuhi semua orang.

6.4. TEKNIK PENULISAN UNTUK PEMBACA NON TEKNIS

Untuk menulis FS yang baik memang sulit sekali. Jika FS menjelaskan sebuah sistem teknis, maka disebut dokumen teknis, tetapi FS ini ditulis untuk pembaca non teknis. Tulislah dari sudut pandang user – gunakan terminologinya. Untuk itu anda harus mempelajari bisnis user dan bahasanya.

Alasan terbesar yang menyebabkan kesalah pengertian dokumen adalah kata-kata yang memiliki dua arti. Hal yang sama, hindari janji-janji yang sulit untuk dilakukan.

6.5. KEGUNAAN LAIN UNTUK SPESIFIKASI FUNGSI (*OTHER USES FOR THE FUNCTIONAL SPECIFICATION*)

FS yang baik dapat digunakan untuk :

- memperkenalkan proyek kepada anggota Tim proyek yang baru.
- User dapat menggunakannya untuk memperkenalkan sistem yang baru ke pihak manajemen, atau ke bagian-bagian lain.
- Tetapi yang paling penting adalah bagian-bagian yang menjelaskan *menu*, *form*, *query*, dan *report* dapat digunakan dalam *User's Guide*.

6.6. CASE SOFTWARE TOOLS UNTUK ANALISIS

Computer Aided Software Engineering (CASE) digunakan sebagai suatu paket *software tools* pada masing-masing fase dari daur hidup sistem. Terdapat beberapa produk software yang mutunya bagus yang membantu anda untuk melakukan analisis.

Contohnya :

1. **Excelerator**. *Excelerator* dapat digunakan untuk menggambarkan *Data Flow Diagram* (DFD) tingkat tinggi, seperti pada gambar 6.2., kemudian memecah DFD ke level-level berikutnya yang lebih rendah.
2. Pada mini komputer, alat seperti **DECDESIGN** mendukung fase analisis dengan menggambarkan DFD atau *Entity Relationship Diagram* (ERD), seperti pada saat menggambar *Structure Chart* dan *Diagram State Transition*.

6.7. MENINJAU KEMBALI PERENCANAAN (REVISING THE PLAN)

Perencanaan adalah proses pengulangan. Lakukan perbaikan PPP segera setelah analisis dilakukan. Apakah tugas-tugas masih dapat diperkirakan, ditentukan, dijadwalkan, dan diselesaikan? Yang paling penting adalah tanyakan apakah sumber daya yang diperlukan untuk masing-masing tugas masih tersedia ketika dibutuhkan?

Berikut ini adalah daftar pendek dari masalah-masalah yang dapat terjadi dalam tiga fase berikutnya (*Design, Programming, System Test*), selama pelaksanaan rencana berikutnya:

- *Programmer* kunci atau perancang mengundurkan diri.
- Komputer pengembangan tidak tersedia.
- Peralatan hardware yang khusus tidak ada / terwujud tepat pada waktu dibutuhkan.
- Paket software dengan *release* terbaru (atau hardware) tidak bekerja.
- Sumber daya yang disediakan oleh pihak ketiga tidak terwujud.

Rencana Pelatihan Untuk Anggota Proyek (*Training Plans For The Project Members*)

Ketika akhirnya staf telah diputuskan, lakukan pemeriksaan untuk melihat siapa-siapa saja yang membutuhkan pelatihan. *Programmer* anda merupakan calon yang paling memungkinkan. Jadwalkan semua pelatihan yang akan dilakukan pada akhir disain.

Diharapkan FS dinegosiasikan atau ditinjau kembali; jadwalkan waktu untuk persetujuan dan perbaikan. Atur batas akhir penyelesaian. Jika tidak disetujui diantara individu-individu atau departemen-departemen menyebabkan '*analysis paralysis*', ambil satu orang dari tiap departemen dan kumpulkan dalam satu ruang dan tekankan untuk tidak menunda pertemuan sampai masalah terpecahkan.

Dan yang terakhir, kita tinjau kembali kejadian-kejadian utama dalam fase analisis:

1. Spesifikasi Fungsi (FS) yang disetujui dan ditandatangani oleh kedua belah pihak.
2. Jika kedua langkah proposal digunakan, *Development Proposal* telah ditulis dan dibeli oleh user.
3. PPP diperbaiki untuk memasukkan perhitungan-perhitungan baru dan jadwal-jadwal; sumber-sumber masih dijalankan untuk seluruh kegiatan.
4. Disain tingkat atas (*The Top Level design / TLD*) telah dilakukan. Hal ini mungkin tidak jelas, tetapi anda harus mengerjakan TLD ketika anda menemukan gagasan dan menggambarkan gambar 6.1. Ini mungkin bukan TLD terbaik, meskipun pada akhirnya akan digunakan juga, tetapi itu merupakan terobosan pertama bagaimana sistem akan bekerja dan bagian utama yang akan diproduksi.