

# **BAB I**

## **KEBUTUHAN DOKUMENTASI**

### **TUJUAN**

- Memahami kebutuhan dokumentasi
- Mengetahui parameter-parameter kebutuhan dokumentasi
- Mengetahui tahapan proses dokumentasi Proyek

### **1.1 Kebutuhan Dokumentasi**

Dokumentasi menjadi salah satu kebutuhan manajemen pengembangan sistem/software. Fungsi dokumentasi adalah merekam kebutuhan *user* yang akan diimplementasikan dalam bentuk spesifikasi dan instruksi.

Sejumlah proyek pengembangan software, terlepas dari aplikasinya, telah mengembangkan sejumlah dokumentasi yang terkait dengan proyek tersebut. Untuk sistem ukuran menengah, dokumentasinya mungkin akan mengisi beberapa lemari arsip; untuk sistem yang lebih besar, mungkin ini akan mengisi beberapa ruangan. Proporsi tingginya biaya proses software disertakan dalam menghasilkan dokumentasi ini. Lebih lanjut, kesalahan dan kelalaian dokumentasi dapat menimbulkan kesalahan pada pengguna akhir dan konsekuensinya menimbulkan tambahan biaya dan gangguan terkait dengan software tersebut. Oleh karena itu, manajer dan teknisi software harus memperhatikan kebutuhan dokumentasi dan biaya terkait pengembangan software itu sendiri.

Dokumen-dokumen yang terkait dengan proyek software dan sistemnya telah dikembangkan dengan beberapa persyaratan seperti tersebut di bawah ini:

1. Dokumen harus bertindak sebagai media komunikasi antar anggota dari tim pengembang.
2. Dokumen harus menjadi *repository* (gudang informasi) sistem yang dapat digunakan oleh teknisi perawatan.
3. Dokumen seharusnya menyediakan informasi bagi manajemen untuk membantu mereka merencanakan dana dan menjadwalkan proses pengembangan software.

4. Beberapa dokumen harus memberikan informasi kepada pengguna bagaimana menggunakan dan mengatur sistem.

Untuk memenuhi persyaratan tersebut di atas diperlukan berbagai tipe dokumen, dari dokumen kerja informal hingga manual pengguna yang dihasilkan secara profesional. Teknisi software biasanya mempunyai tanggung jawab untuk menghasilkan dokumentasi melalui penulis teknis profesional yang membantu melalui prosedur pemolehan eksternal dari informasi yang dikeluarkan.

Salah satu bagian pada sistem informasi adalah untuk menaikkan kualitas proyek yang biasanya melibatkan elemen berikut ini :

- a) Metodologi

Suatu cara atau metode untuk mencapai tujuan. Suatu metodologi berlaku secara umum, dengan perencanaan tingkat tinggi dan digunakan sebagai landasan setiap proyek. Ada beberapa metode khusus untuk beberapa jenis proyek yang khusus, seperti metodologi untuk Internet atau Intranet.

- b) Dokumentasi

Dokumen khusus, yang pada awal proyek akan menerangkan secara garis besar. Yang mana hal tersebut akan dilengkapi pada setiap proyek yang dilaksanakan. Contoh dokumentasi adalah : *Functional Specification, Cost-benefit Analysis and Return of Investment*.

- c) Standar

Panduan yang disusun dan digunakan pada suatu institusi untuk menyelesaikan suatu pekerjaan. Contoh standar ini adalah; kesepakatan penamaan untuk berbagai macam kode, kesepakatan layar GUI, kesepakatan data modelling. Standar ini penting karena merupakan landasan pengembangan sebagai kerangka kerja komunikasi. Juga untuk mengontrol kualitas serta menjaga kontinuitas pengembangan.

Hal penting dari suatu *Bussines Process Reengineering* (BPR) adalah mengoptimasi metode untuk melaksanakan suatu pekerjaan, yang dapat dipakai ulang untuk proyek mendatang. Setelah suatu metodologi ditentukan, akan lebih mudah untuk para manajer memahami pola antar proyek dan menentukan mana yang bekerja dan mana yang tidak. Akan lebih mudah mengetahui mengapa suatu proyek menjadi gagal, dan mengetahui titik penting yang mengakibatkan

kesuksesan suatu proyek. Kesuksesan inilah yang sering disebut dengan *best practices*. Bila hal ini telah didefinisikan maka dapat digunakan berulang-ulang sehingga dikenal dengan istilah *reusable process* yaitu desain software yang bersifat adaptif dan akomodatif terhadap perubahan keperluan software oleh user.

Untuk kesuksesan suatu proyek, desain dan implementasi harus dilakukan dengan sama baiknya. Pada suatu BPR, problem sering kali didefinisikan oleh manajer tingkat tinggi, yang tak mengetahui bagaimana pekerja sebenarnya mengerjakan hal tersebut. Pada kasus lain, konsultan sering melakukan proses BPR tanpa memahami dengan benar bagaimana proses tersebut dilaksanakan. Solusi dari permasalahan ini adalah kesepakatan penggunaan BPR dan keterlibatan setiap personal. Proses harus dievaluasi oleh developer, analis, arsitek sistem, manajer, dan juga oleh pengguna. Pendekatan ini lebih kepada pendekatan *bottom up* daripada *top down*. BPR membutuhkan sejumlah konsensus yang harus disepakati untuk dilaksanakan secara bersama-sama.

## **1.2 Dokumentasi Proyek**

Dokumentasi adalah hal pertama yang harus ditentukan dan diselesaikan. Hal yang penting agar dokumentasi dapat disusun dengan sukses adalah dilakukan dengan cara mengintegrasikan dokumentasi ini dengan metodologi, sehingga proses dokumentasi dilakukan pada setiap langkah *development* dilakukan, daripada melakukannya setelah selesai proyek. Bentuk dasar dari dokumentasi ini sebaiknya juga dilakukan untuk proyek-proyek yang lainnya. Pada suatu proyek biasanya terdapat enam proses yang saling terkait dan dinamis, yaitu:

1. Pendefinisian
2. Perencanaan
3. Organisasi
4. Pengawasan
5. Penyelesaian
6. *Leading*

Setiap proses akan memiliki keluaran yang akan menjadi masukan bagi proses yang lainnya. Proses-proses ini memberikan beberapa keuntungan termasuk :

- Mengetahui dampak teknologi dan bisnis
- Menghitung estimasi biaya sesungguhnya
- Menentukan tingkat usaha
- Mencapai suatu penyelesaian yang paling efektif biayanya
- Memilih perangkat bantu dan teknik terbaik

### 1.2.1 Pendefinisian

Dengan mendefinisikan proyek dengan tetap, diharapkan proyek dapat dimulai dan diakhiri dengan biaya yang paling efektif. Termasuk menjawab; *who, what, when, where, why and how* dari pelaksanaan proyek tersebut. Perangkat bantu untuk melaksanakan tugas ini disebut dengan *Statement of the Works* (SOW). SOW adalah kesepakatan antar *client* dan *developer*. Dokumen ini ditulis berdasarkan perspektif bisnis dan teknis yang berisi topik-topik berikut ini :

- Pengantar (misal informasi latar belakang)
- Tujuan dan obyektif (misal; *cost*, jadwal dan kualitas)
- *Scope* (misal; aplikasi HTML atau VRML)
- Assumsi (misal; kemampuan penanganan peningkatan traffic jaringan)
- *User*
- Sumber daya (misal spesialis jaringan, programmer)
- *Milestone* untuk penjadwalan (misal waktu akhir testing)
- Pembiayaan (biaya langsung dan *overhead*)
- Amandement (definisi ulang dari penyerahan pekerjaan)
- Tanda tangan (manajemen senior dan komunitas pengguna)

SOW memberikan keuntungan ketika digunakan untuk memulai suatu proyek, yaitu :

- Menjelaskan biaya dan jadwal juga asumsi utama tentang proyek
- Menjelaskan peranan dan tanggung jawab
- Mengukuhkan definisi hal yang akan dicapai proyek Intranet tersebut
- Mendorong diselesaikannya proyek tersebut, karena adanya kesepakatan tertulis dalam dokumen tersebut (tanda tangan).

Selain itu, SOW ini akan membantu menentukan tanggung jawab sekuriti pada tingkat tinggi, perawatan dokumentasi, perangkat lunak, data, perangkat

keras, dan pengelolaan sistem. Dengan kata lain akan mendefinisikan siapa yang berperan sebagai *web-masters*, *document-master*, dan *document-owners*. SOW juga mencegah permasalahan yang timbul di tahapan berikutnya dari pengembangan sistem.

### **1.2.2 Perencanaan**

SOW menjabarkan biaya secara kasar, penjadwalan, kualitas, dan sumberdaya manusia pada suatu proyek. Dengan informasi ini perencanaan dilakukan dengan berdasarkan pada informasi ini. Perencanaan sebagai langkah berikutnya meliputi 6 tahapan yang dapat dilaksanakan secara berurutan ataupun paralel:

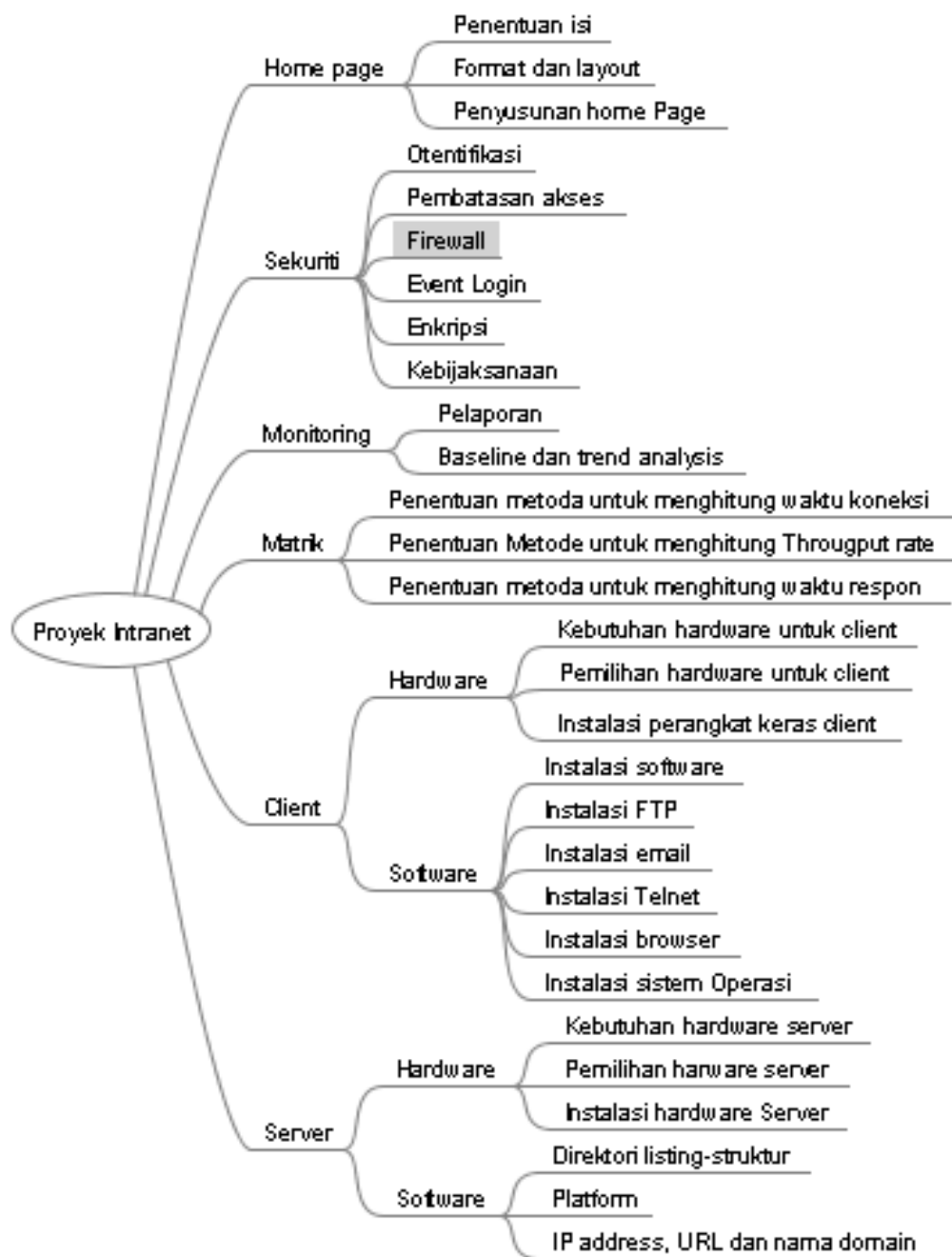
- Menyusun *Work Breakdown Structure* (WBS)
- Mengestimasi waktu pelaksanaan proyek
- Mengalokasikan sumber daya
- Menghitung pembiayaan
- Menyusun jadwal kerja
- Pengelolaan resiko

#### **Menyusun WBS**

Pada dasarnya WBS merupakan suatu daftar yang bersifat *top down* dan secara hirarkis menerangkan komponen komponen yang harus dibangun dan pekerjaan yang berkaitan dengannya. Sebagai contoh, WBS untuk proyek Intranet seperti pada gambar 1.1.

Model WBS memberikan beberapa keuntungan:

- Memberikan daftar pekerjaan yang harus diselesaikan
- Memberikan dasar untuk mengestimasi, mengalokasikan sumber daya, menyusun jadwal dan menghitung biaya
- Mendorong untuk mempertimbangkan secara lebih serius sebelum membangun suatu proyek Intranet.



Gambar 1.1. WBS untuk proyek Intranet

### Mengestimasi Waktu Pelaksanaan Proyek

Dengan memanfaatkan daftar pekerjaan pada WBS, dapat dilakukan pekerjaan dengan memperkirakan waktu yang dibutuhkan untuk menyelesaikan setiap pekerjaan tersebut. Perkiraan dilakukan dengan beberapa pertimbangan,

yaitu ketersediaan sumber daya dan kompleksitas yang kemudian dijabarkan dalam kalender. Biasanya optimasi dilakukan secara:

a) ***Most Optimistic***

Waktu ideal untuk menyelesaikan pekerjaan, diasumsikan segala sesuatunya berjalan lancar, dan sempurna.

b) ***Most Likely***

Waktu yang dibutuhkan pada kondisi kebanyakan, tipikal dan normal.

c) ***Most Pessimistic***

Waktu yang dibutuhkan ketika keadaan paling sulit terjadi.

Estimasi waktu dilakukan dan dibagi dalam bentuk unit (misal 8 jam per hari). Estimasi waktu untuk suatu proyek Intranet lebih sulit dari proyek pengembangan aplikasi lainnya. Hal ini karena masih sedikit proyek yang dapat digunakan sebagai patokan menghitung waktu pelaksanaan. Dalam mengestimasi waktu ini juga harus dipertimbangkan beberapa hal, misal pengalaman teknologi server yang digunakan, keahlian Perl, CGI, Java dan HTML, browser, dan juga bekerja dalam lingkungan TCP/IP.

### **Penentuan Resiko**

Prioritas penting ditentukan pada seetiap proyek, termasuk juga pada proyek Intranet. Sebab seperti halnya Internet ada beberapa permasalahan sekuriti (seperti akses tanpa hak), dan karena adanya banyak komponen pembentuk sistem (misal browser dan server) yang terlibat, resiko dapat menjadi tinggi. Penentuan resiko akan membantu melakukan identifikasi resiko yang dihadapi setiap komponen. Dengan informasi ini seorang manajer proyek dapat menentukan tingkat kepentingan setiap tugas dan menentukan estimasi waktu untuk itu. Manajer proyek dapat berkonsentrasi pada waktu dan sumber daya pada elemen yang terkritis dari penjadwalan.

### **Menyusun Jadwal Kerja**

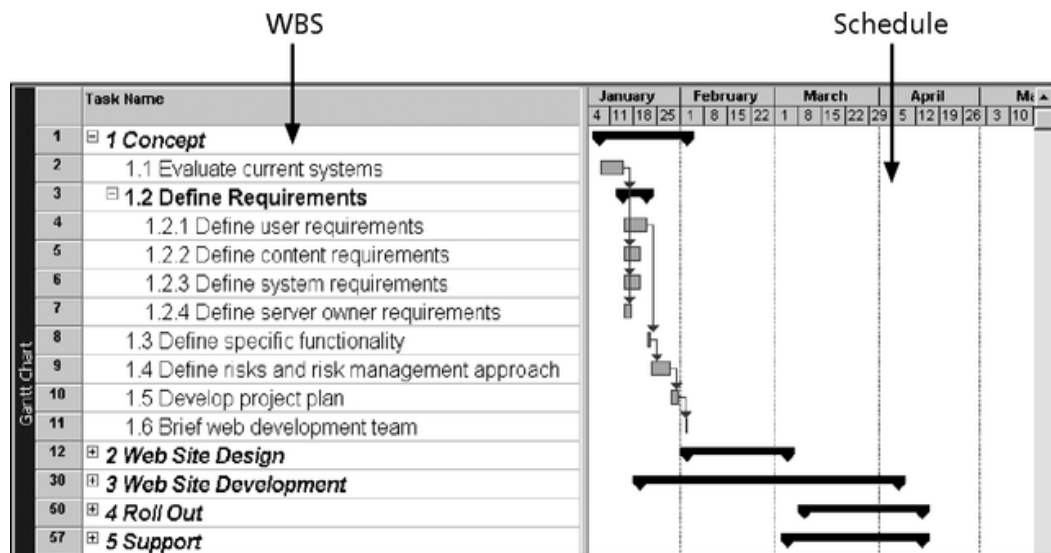
Pada dasarnya ada dua jenis model deskripsi penjadwalan :

- **Bar Chart**, yang hanya menerangkan ow time dari setiap pekerjaan dan tanpa keterkaitan antar pekerjaan. Deskripsi ini paling baik digunakan

pada presentasi

- **Network diagram**, yang menunjukkan keterkaitan antar tugas dan mengidentifikasi saat kritis pada jadwal.

Suatu network diagram, merupakan cara terbaik untuk merencanakan secara detail, dan mengikuti perkembangan proyek. Diagram ini akan menghubungkan pekerjaan terkait, dan waktu mulai dan berakhirnya dari pekerjaan tersebut. Mengidentifikasi keterkaitan pekerjaan pada proyek Intranet adalah sangat penting sebab komponen-komponen tersebut saling terkait agar dapat bekerja sesuai dengan fungsinya. Berikut adalah contoh jadwal untuk proyek intranet:



Gambar 1.2 Pembuatan jadwal dengan menggunakan Ms Project

## Mengalokasikan Sumber Daya

Pada dasarnya pengaturan waktu setiap pekerjaan harus dilakukan agar ketersediaan dan kemampuan sumber daya bisa seimbang. Level load harus ditentukan dari sumber daya agar tak ada personal yang bekerja terlalu berat dan yang terlalu ringan. Pada proyek Intranet hal ini sulit, karena tidak tersedianya sumberdaya manusia yang memiliki keahlian tersebut, oleh sebab itu harus disusun jadwal yang realistis. Dan bahkan mungkin dilakukan revisi penjadwalan.



## Menghitung Pembiayaan

Yang menjadi permasalahan, apakah biaya yang akan dikeluarkan sesuai dengan SOW. Jika sesuai, maka pekerjaan perencanaan selesai, bila tidak harus dilakukan revisi. Bila memang sulit harus dilakukan negosiasi dengan pihak pemberi kerja. Ketika melakukan perhitungan biaya perlu dipertimbangkan beberapa biaya tersembunyi, misal training, dokumentasi.

### 1.2.3 Organisasi

Proses ini adalah proses yang melibatkan penyusunan suatu infrastruktur yang akan memaksimalkan efisiensi dan efektifitas ketika melaksanakan proyek. Yang harus dipertimbangkan adalah :

- 1) Struktur tim
- 2) Dokumentasi
- 3) Pertemuan

#### Struktur Tim

Ditentukan dengan menjelaskan tentang pembagian peranan, tanggung jawab, dan yang berhubungan dengan pelaporan. SOW sebaiknya menyediakan dasar untuk menjelaskan peranan utama, tanggung jawab, dan hubungan pelaporan. Informasi ini membantu untuk mempersiapkan struktur team, seperti untuk menghasilkan diagram organisasi dan matriks tanggung jawab.

#### Dokumentasi

Dokumentasi adalah hal yang penting sekali, sebab *user* memiliki peranan penting dalam membuat dan merawat kandungan software. Diagram arsitektur, perangkat bantu *mapping*, dan manual on line merupakan perangkat bantu dokumentasi teknis.

Pertemuan terdiri dari 3 jenis :

- *Status review meeting*, dilakukan secara regular untuk mengumpulkan informasi mengenai kondisi dari pekerjaan individu.
- *Checkpoint review meeting*, dilakukan untuk mencapai milestone besar, seperti men setup server.

- *Staff meeting*, dilakukan untuk bertukar informasi dan bertukar pengalaman bagi seluruh pihak yang terlibat

Pihak yang menghadiri pertemuan ini dapat bervariasi, tapi minimal pihak pengguna harus ada yang diundang. Ini menyebabkan mereka tidak saja merasa terlibat tetapi juga memperoleh informasi mengenai sekuriti, hak akses, dan kandungan dokumen. Hal ini akan mendorong dapat diselesaikannya proyek ini.

#### **1.2.4 Pengawasan**

Proses ini menjamin bahwa proyek Intranet efektif pembiayaannya dan sesuai dengan yang direncanakan. Proses ini terdiri dari :

- *Status collection*
- *Change control*
- *Corrective action*

##### *Status collection and Assesment*

Proses ini akan mengumpulkan data tentang penyelesaian suatu pekerjaan atau pencapaian suatu milestone. Kemudian membuat penilaian mengenai perkembangan yang dilakukan. Proses ini memiliki sisi bisnis dan teknis. Sisi teknis melibatkan penilaian kualitas pekerjaan yang dilakukan misalnya bagaimana HTML dan CGI yang disusun. Pada sisi bisnis meliputi tingkatan mana pekerjaan itu dilakukan berdasarkan waktu tertentu.

##### *Change control*

Proses ini melibatkan pekerjaan mengevaluasi pelaksanaan teknis dan jadwal. Dalam pelaksanaan membutuhkan jawaban akan pertanyaan seperti :

- Apakah sebenarnya perubahan yang terjadi (misalnya arsitektur jaringan).
- Apa dampaknya bagi finansial, jadwal dan kualitas sistem.
- Bagaimana penanganan perubahan tersebut, misalnya terhadap *user* dan komunitas sistem informasi.
- Bilamana perubahan tersebut akan menyebabkan suatu efek, misal setelah Intranet terpasang dan berjalan.

### *Corrective action*

Langkah ini melakukan revisi pendekatan yang dilakukan untuk mencapai tujuan proyek sesuai dengan SOW dan perencanaan. Langkah ini berkaitan sekali dengan langkah status collection and assesment, sebab langkah yang dibutuhkan misal perencanaan ulang, bergantung apakah *corrective action* ini perlu dilakukan secara besar atau cukup sedikit saja.

#### **1.2.5 Penyelesaian proyek**

Proses yang terlibat dalam proyek ini adalah melakukan pengumpulan dan analisis data dan melakukan transisi yang baik dari proses pengembangan ke implementasi. Keluaran utama dari proses ini adalah hal yang dipelajari selama pelaksanaan proyek - *lesson learned document*. Dokumen ini mengidentifikasi apa yang dilakukan dengan baik dan apa yang tak berhasil dilakukan. Hal itu berdasarkan data yang dikoleksi mengenai unjuk kerja proyek melalui kumpulan hasil statistik, wawancara dan review setelah implementasi. Dokumen ini berguna bagi organisasi besar yang mungkin akan melakukan pemasangan site Intranet yang berjumlah banyak. Pengalaman yang diperoleh dari proyek pertama ini akan memberikan pandangan bagi manajer proyek untuk proyek mendatang.

Suatu hal yang penting lagi adalah bagaimana hasil dari proyek ini. Tendensi apakah yang terjadi di antara personal yang terlibat pada pengembangan proyek pada saat mendekati akhir proyek. Bila suatu proyek akan selesai biasanya anggota tim menjadi menurun produktifitasnya. Oleh karena itu, sebaiknya bila seorang anggota tim telah melakukan suatu tugas berat, sebaiknya segera dibebaskan bila memang telah tidak ada pekerjaannya lagi. Ini menyebabkan personal tersebut dapat bertugas di proyek Intranet yang lainnya lagi.

#### **1.2.6 Leading**

Tahapan ini penting sekali hanya akan terjadi bila kelima proses sebelumnya dilakukan dengan benar. Pada tahapan ini dibutuhkan pembentukan suatu lingkungan kerja yang mendorong pihak yang terlibat, sehingga dapat tercapainya tujuan. Untuk mencapai hal tersebut, manajer proyek haruslah:

- Membuat visi yang jelas bagi proyek

- Berkomunikasi dengan efektif
- Menjaga motivasi yang tinggi
- Menjaga fokus dari visi
- Menyediakan lingkungan yang mendukung
- Mendorong penyusunan tim.

Beberapa langkah tersebut sulit dilaksanakan karena biasanya manajer proyek tidak terlalu memiliki kendali dalam penggunaan sumber daya. Hal ini menjadi lebih rumit untuk proyek Intranet yang melibatkan banyak pihak dengan keahlian terbatas, orientasi fungsi yang tak jelas. *Web master* dan *document owner* bukanlah nama pekerjaan yang unik tapi juga membutuhkan keahlian khusus.

Suatu proyek akan dapat dilakukan dengan baik bila telah dilakukan proses *engineering* yang baik. Ini berlaku baik untuk pengembangan program dengan produk jadi atau dengan kontraktor atau juga dengan tim sendiri. Akan lebih baik menghabiskan waktu lama untuk melakukan *desain* dan penataan awal yang baik daripada terburu-buru melakukan implementasi. Sehingga sudah sewajarnya dilakukan standardisasi dan penggunaan dokumentasi yang baik. Biasanya suatu tim pengembang sistem informasi cenderung untuk meninggalkan metodologi yang standar dengan alasan keterbatasan waktu.

## **PERTANYAAN DAN TUGAS**

1. Apa yang dimaksud dengan kebutuhan dokumentasi?
2. Apa fungsi dokumentasi dalam proyek software?
3. Hal-hal apa saja yang bias dilakukan untuk mendukung peningkatan kualitas suatu Proyek
4. Sebut dan jelaskan secara ringkas tahapan dalam proses dokumentasi proyek

## **BAB II**

### **DOKUMENTASI PERANCANGAN SISTEM**

#### **TUJUAN**

- Mengetahui bagaimana mendokumentasikan perancangan sistem
- Mengetahui aktivitas pembuatan dokumentasi perancangan sistem
- Mengetahui macam-macam dokumentasi perancangan sistem

Untuk proyek software yang besar, biasanya dokumentasi dilaksanakan sebelum proses pengembangan dimulai. Proposal untuk mengembangkan sistem dapat dibuat sebagai respon terhadap permintaan untuk tender oleh klien eksternal atau untuk merespon dokumen strategi bisnis lainnya. Untuk beberapa tipe sistem, suatu dokumen persyaratan yang menyeluruh dapat dibuat, yang mana mendefinisikan fitur yang dibutuhkan dan perilaku sistem yang diharapkan. Selama proses pengembangan itu sendiri, semua urutan dokumen yang berbeda dapat dihasilkan, diantaranya perencanaan proyek, spesifikasi rancangan, rencana uji coba, dan sebagainya. Aktivitas Perencanaan Pengembangan perangkat lunak diantaranya:

- Memilih suatu model untuk proses pengembangan
  - Mengontrak dan menyusun tim pembuat software
  - Membeli atau menyewa perangkat keras/lunak yang dibutuhkan
  - Memilih produk berpotensi berdasarkan pengalaman keorganisasian, sumber daya, dan tujuan.
  - Mengevaluasi informasi pasar mengenai viabilitas produk
  - Mengestimasi biaya, jadwal, resiko dan harga dari produk
  - Memilih bentuk laporan dan cara pengukuran perkembangan proyek
  - Memilih notasi untuk spesifikasi dan perancangan
  - Memilih bahasa pemrograman
  - Meletakkan standar untuk dokumentasi, coding, verifikasi dan pengujian
  - Memilih mekanisme pengaturan konfigurasi
  - Menyiapkan bahan dan personil untuk instalasi
- Beberapa dokumen yang biasa digunakan adalah :
- Project plan
  - Spesifikasi *desain*

## 2.1 Project plan

Suatu *project plan* (perencanaan proyek) berisi:

### 1. Pengantar, berisi:

- Deskripsi permasalahan
- Deskripsi lingkungan masalah
- Tujuan client, dan organisasi serta sistem.
- Solusi yang diajukan dan ruang lingkupnya

### 2. Proposal

- Fungsi yang diberikan pada solusi yang diajukan
- Strategi umum untuk mengembangkan solusi
- Peran pengguna dan perangkat keras pada solusi tersebut
- Keuntungan dan kerugian solusi tersebut

### 3. Keterbatasan sistem (constraint)

- Prioritas kustomer
- Profil pengguna
- Usia pengharapan dari produk
- Pra-syarat keandalan (reliabilitas)
- Pra-syarat kinerja
- Lingkungan perangkat keras dan antar muka yang telah ada
- Pengembangan mendatang dari produk
- Pra-syarat, bahasa pemrograman untuk implementasi (jika ada)
- Pra-syarat pelatihan, instalasi dan dokumentasi.
- Ketersediaan pada lingkungan pengguna
- Solusi alternatif
- Studi feasibilitas

### 4. Estimasi

- Jadwal
- Staf dan organisasi
- Budget
- Analisis Cost/Benefit
- Analisis resiko
- Dokumen yang diberikan

- Perangkat lunak yang dibutuhkan
- Fasilitas dan perangkat keras yang dibutuhkan

## 5. Prosedur

- Model proses
- Metodologi dan notasi
- Standardisasi dan jaminan kualitas
- Accountability monitoring
- Kendali produk
- Data pengujian dan sumber data
- Kriteria akseptansi dan metoda pembayaran

## 6. Referensi

- Dokumentasi yang digunakan dalam pengembangan
- Kamus istilah
- Kontrak yang diusulkan (jika ada)

## 2.2 Spesifikasi desain

Dokumen ini pada dasarnya menerangkan tentang kebutuhan sistem yang akan dibuat. Beberapa paradigma perancangan akan menentukan model desain dan notasi. Beberapa pendekatan desain adalah :

- *Access-oriented design*
- *Data-structure-oriented design*
- *Data ow design*
- *Functional design*
- *Imperative design*
- *Object-oriented design*
- *Parallel design*
- *Real-time design*
- *Rules-oriented design*
- *User centered design*

Suatu dokumen spesifikasi desain terdiri dari :

### 1. Pendahuluan

- Garis besar permasalahan

- Lingkungan aplikasi dan karakteristik pengguna
- Notasi yang digunakan dalam *desain*
- Tujuan proyek

## 2. Spesifikasi secara singkat

- Fungsi perangkat lunak
- Teknik yang digunakan untuk menyelesaikan masalah ini terutama ketika desainer tidak memiliki pengetahuan khusus
- Kinerja yang harus dicapai
- Deskripsi data
- Hubungan data
- Prioritas implementasi
- Spesifikasi real time
- Spesifikasi interaksi manusia dan mesin yang digunakan.
- Batasan
- Eksepsi
- Modifikasi dan perawatan yang diprediksi

## 3. Desain arsitektur

- Modul hirarki dan diagram interface
- Deskripsi fungsi dan data
- Spesifikasi interface

## 4. Desain secara detail untuk tiap modul :

- Deskripsi modul dan spesifikasi interface
- Deskripsi proses
- Definisi struktur data
- Pra-syarat inisialisasi
- Spesifikasi penanganan eksepsi
- Alternatif desain, untuk tiap desain yang ditolak disertakan keterangan alasan penolakan serta kondisi yang menyebabkan desain yang terpilih.

## 5. Referensi

- Dokumentasi yang digunakan untuk mengembangkan *desain*
- Daftar terminologi



## **PERTANYAAN DAN TUGAS**

1. Apa yang dimaksud dengan dokumentasi perancangan sistem?
2. Bagaimana cara mendokumentasikan perancangan sistem?
3. Aktivitas apa saja dalam pembuatan dokumentasi perancangan sistem?
4. Ada berapa hal yang bisa dilakukan untuk dokumentasi perancangan sistem?

## **BAB III**

### **CONTOH MODEL DOKUMENTASI PERANCANGAN SISTEM**

#### **TUJUAN**

- Mengetahui contoh model dokumentasi perancangan sistem
- Mengetahui tahapan pembuatan dokumentasi perancangan sistem

Berikut ini diberikan suatu contoh dokumentasi perancangan suatu sistem komputer yang berisi :

- Bagian 1. Kebutuhan pengguna
- Bagian 2. Spesifikasi
- Bagian 3. Desain
- Bagian 4. Implementasi dan pemilihan teknologi
- Bagian 5. Pengujian
- Bagian 6. Aplikasi (bila perlu)
- Lampiran

#### **3.1 Kebutuhan Pengguna (*user requirement*)**

Bagian ini menjelaskan kebutuhan dari sistem baik, dari sisi pengguna, fungsi maupun teknologi. Bagian ini terdiri dari definisi kebutuhan pengguna, analisis kebutuhan, model kebutuhan.

##### **3.1.1 Definisi Kebutuhan (*Requirement definition*)**

Bagian ini mendefinisikan kebutuhan dan hal-hal yang berkaitan dengan kebutuhan akan sistem secara menyeluruh. Biasanya dalam kasus kondisi sebenarnya, dokumen ini harus disetujui oleh pemberi kerja dan pelaksana kerja. Bagian yang dibahas dalam dokumen ini adalah:

- ***Purposeful requirement.***  
Menjelaskan mengapa kebutuhan itu perlu dipenuhi
- ***Functional requirement.***  
Apakah fungsi sebenarnya yang dibutuhkan oleh *user* dari sistem ini.

➤ ***Nonfunctional requirement.***

Dan kebutuhan sistem yang tidak berkaitan dengan fungsinya yang melibatkan seperti, bisa dimodifikasi, testing, dan portabilitas

➤ ***User Profile.***

Menerangkan tentang pengguna dari sistem ini, terutama yang berkaitan dengan tingkat pengenalan terhadap teknologi yang akan diterapkan.

### **3.1.2 Analisis Kebutuhan (*Requirement Analysis* )**

Bagian ini akan menganalisis kebutuhan yang dijabarkan pada bagian sebelumnya serta kaitannya dengan hal-hal yang mempengaruhi pelaksanaan ataupun pemilihan solusi.

➤ ***Requirement Prioritisation.***

Menjabarkan kebutuhan mana yang terpenting dari kebutuhan-kebutuhan yang timbul untuk sistem yang akan dibuat tersebut.

➤ ***Constraint and Risk Analysis.***

Memahami halangan dan resiko yang ada untuk mengimplementasikan model dan mengaplikasikan sistem ini.

➤ ***Trade-off analysis.***

Mengetahui hal-hal yang saling bertentangan dalam perancangan, pengimplementasian serta pengaplikasian dari solusi.

### **3.1.3 Model Kebutuhan (*Requirement model* )**

Bagian ini menyusun model kebutuhan tersebut menjadi suatu bentuk model kebutuhan. Disusun atas *hierarchical (functional)* model dari prioritas, *risk functional*, *non-functional requirement* dan suatu prototype yang interactive. Model ini digunakan untuk mengurangi ketidakpastian kebutuhan sistem, sehingga tercapai kesepakatan antara pemberi kerja dan pelaksana kerja.

➤ ***Functional model***

➤ ***Exploratory prototype***

➤ ***Requirement trace***

➤ ***The conceptual model***

### 3.2 Spesifikasi

Pada bagian ini dijabarkan spesifikasi detail dari sistem yang harus dibuat, spesifikasi ini meliputi hal-hal sebagai berikut: spesifikasi siklus operasi sistem, spesifikasi fungsional, komponen sistem dan spesifikasi kinerja.

#### 3.2.1 Spesifikasi Siklus Operasi Sistem (*System Operating Cycle Specification*)

Dalam bagian ini dijelaskan siklus penggunaan sistem ini, tujuan untuk mendapatkan spesifikasi teknis terutama akan berguna pada bagian implementasi. Sistem *fault tolerant* 24 jam akan berbeda dengan sistem yang beroperasi 1 jam sehari. Begitu juga siklus kerja sistem, atau langkah kerja satu demi satu dispesifikasikan.

#### 3.2.2 Spesifikasi Fungsional

Pada bagian ini dijelaskan fungsi dari sistem yang akan dibuat.

➤ *Essential Capabilities*

Dijabarkan fungsi utama dari sistem. Ini merupakan fungsi minimal yang harus dipenuhi oleh sistem yang dibuat.

➤ *Additional Capabilities*

Dideskripsikan fungsi tambahan yang timbul karena dipenuhi dan digunakan sistem ini.

➤ *Future Capabilities*

Menjelaskan fungsi pada masa datang dari sistem ini. Penjelasan ini erat kaitannya dengan *marketing strategy*, *technological* dan *Sales*.

#### 3.2.3 Komponen Sistem

Pada bagian ini dijabarkan komponen-komponen yang dibutuhkan untuk sistem secara keseluruhan. Termasuk *software*, *hardware*, *user*, dan organisasi penunjang. Untuk mempermudah penjabaran pembagian tugas antar komponen dapat dilakukan *decomposition process* dan penjabaran *relationship* antar komponen.

### 3.2.4 Spesifikasi kinerja

Pada bagian ini dijelaskan performansi / unjuk kerja yang ingin dicapai serta kemungkinan keterbatasan di dalam penggunaan sistem. Setiap kebutuhan diusahakan dispesifikasikan dengan jelas, termasuk karakteristik elektrik dan fisis.

➤ *Characteristics and Constraints*

Merupakan karakteristik tiap komponen pendukung sistem, termasuk karakteristik fisis, elektrik, maupun karakteristik perangkat lunak (misal; *user friendly* dan interaktif). Juga dijabarkan keterbatasan di dalam penggunaan sistem (misal; pengguna harus menggunakan kaki sebagai masukan).

➤ *Physical characteristics*

Merupakan penjelasan bentuk luar dari sistem secara keseluruhan, misal; ukuran berat dan lain-lain.

➤ *Environmental characteristics*

Lingkungan tempat kerja dari sistem ini

➤ *Human Factors*

Dijabarkan pengaruh manusia di dalam operasi sistem, baik sebagai penentu operasi, misal dalam *decision making system* atau *supervised control system*. Juga pengaruh manusia dalam menimbulkan ketidakakuratan atau kesalahan sistem.

Untuk mempermudah proses spesifikasi dapat digunakan prototyping, contoh I/O screen dan contoh keluaran. Proses pelaksanaan spesifikasi ini dalam pelaksanaan proyek sesungguhnya biasanya menggunakan metoda formal *Joint Application Design* (JAD).

### 3.3 Desain

Pada bagian ini diterangkan metode dari solusi yang digunakan untuk memenuhi kebutuhan yang telah dispesifikasikan pada bagian terdahulu. Yang dijelaskan dalam bagian ini adalah langkah perancangan dari solusi yang ditawarkan. Sedapat mungkin harus dipisahkan antara perancangan dan implementasi. Pada perancangan diusahakan sedapat mungkin yang dilakukan adalah memodelkan solusi secara logika atau secara algoritmis tanpa terkait erat

dengan teknologi yang digunakan dalam proses implementasi model. Keterkaitan implementasi hanyalah menjadi bagian dari model bukan menjadi dasar desain. Langkah-langkah *desain* akan sangat bergantung pada model dari sistem yang dibuat. Penekanan pada desain ini adalah pada desain sistem utama.

### 3.3.1 Desain Sistem Utama

#### ➤ **Diagram blok**

Sebagai langkah pertama adalah menggambar blok diagram sistem secara keseluruhan dan juga menggambarkan interface antara kedua bagian tersebut. Blok diagram terdiri dari 2 bagian, yaitu dunia di luar sistem dan di dalam sistem.

#### ➤ ***Flow of Control***

Dapat menggunakan *Algorithmic State Machine* (ASM) yang menerangkan kondisi-kondisi yang mungkin dari sistem serta transisi dari satu kondisi ke kondisi lainnya.

#### ➤ ***Representation of Data Flow***

Menggambarkan aliran data serta transformasi yang dialami data tersebut dalam sistem

#### ➤ ***Decomposition into Functions***

Membagi-bagi fungsi secara keseluruhan menjadi sub sistem dengan sub fungsinya. Dapat digunakan Tree Diagram.

#### ➤ ***Relationship among Function***

Dijelaskan hubungan antar sub sistem dengan kaitannya terhadap subfungsinya.

#### ➤ ***Module Specification***

Menggambarkan spesifikasi dari modul sistem.

Dalam memilih diagram atau notasi untuk mendesain sistem perlu diperhatikan beberapa poin :

#### ➤ Suatu alat bantu untuk berfikir secara jelas

Suatu diagram yang baik akan membantu orang untuk memahami ide yang kompleks. Suatu diagram notasi harus direncanakan untuk mempermudah

cara pikir dan komunikasi dan untuk berfikir dengan bantuan komputer. Diagram merupakan alat bantu

- Mudah dimengerti

Suatu diagram harus digunakan bila memiliki pengertian yang telah dikenal. Harus dihindari simbol yang sulit dijelaskan.

- Sebagai alat bantu untuk komunikasi dengan *end-user*

*End user* harus dapat memahami desain dan memberikan masukan kepada desainer. Jadi penggunaan notasi yang membingungkan *end-user* sebaiknya dihindari.

### 3.4 Implementasi dan Pemilihan Teknologi

Pada bagian ini dijelaskan metode dan peralatan yang digunakan untuk mengimplementasikan solusi yang telah diajukan dalam bagian *desain*. Sebaiknya alasan pemilihan teknologi yang digunakan haruslah dijabarkan pada bagian ini. Misal alasan pemilihan suatu jenis komponen perlu diberikan dengan jelas dalam bagian ini, misal mengapa memilih CMOS atau TTL.

Dalam bagian implementasi diterangkan skema atau diagram yang digunakan, baik elektronis, fisis ataupun keterangan program dengan menggunakan metode diagram yang sesuai. Misal untuk rangkaian elektronis menggunakan skema elektronis, sedangkan untuk program menggunakan flow chart dan lain lain.

Sebelum dilakukan pemilihan teknologi atau level alat yang digunakan maka harus dilakukan estimasi terhadap beberapa hal :

- Estimasi waktu untuk mengembangkan sistem
- Estimasi panjangnya program
- Estimasi kebutuhan memori
- Estimasi kecepatan eksekusi

Juga harus dipertimbangkan pembagian antara software dan hardware .

### 3.5 Pengujian (*Testing*)

Bagian ini menunjukkan kerja dari sistem baik untuk masukan yang bersifat normal, ataupun untuk masukan yang di luar ambang normal. Setiap

pengujian dilakukan dokumentasi sebagai bukti otentik kemampuan sistem. Pada dasarnya, disamping testing yang bersifat testing dalam kondisi operasi, akan baik pula bila dilakukan testing berikut ini :

➤ ***Recovery testing***

Pengujian dilakukan untuk mengetahui kemampuan sistem, untuk mengembalikan ke kondisi normal setelah suatu masukan atau kondisi di luar dari yang dispesifikasikan. Untuk sistem-sistem yang bersifat *fault-tolerant*, jenis pengujian ini merupakan suatu kewajiban.

➤ ***Stress testing***

Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menangani beban kerja yang berat, sangat baik untuk mengetahui kemampuan maksimal dari sistem.

➤ ***Security testing***

Testing ini dilakukan untuk jenis-jenis aplikasi yang berkaitan dengan keamanan sistem, baik dari pengguna yang melakukan kesalahan secara sengaja ataupun tidak sengaja.

Dalam melakukan uji coba dapat digunakan beberapa metodologi, diantaranya:

➤ *Use-Case*

➤ *White Box*

➤ *Black Box*

➤ *Loop testing*

Dapat digunakan beberapa satuan untuk menunjukkan hasil kerja sistem, bila dalam perangkat keras telah jelas besaran yang digunakan misal: *frekuensi response, slew rate* dll. Untuk perangkat lunak dapat digunakan: *software metric, cyclomatic complexity*. Pada dasarnya suatu testing akan melakukan :

➤ Verifikasi

Menjamin bahwa sistem benar-benar bekerja sesuai fungsinya.

➤ Validasi

Menjamin bahwa sistem benar-benar memenuhi keinginan pemakai.



## **PERTANYAAN DAN TUGAS**

1. Apa yang dimaksud dengan kebutuhan pengguna?
2. Sebutkan hal-hal yang harus diperhatikan untuk mengetahui kebutuhan pengguna
3. Pertimbangan apa untuk membuat spesifikasi sistem
4. Bagaimana cara membuat desain sistem sebuah proyek software
5. Pada bagian implementasi dan pemilihan teknologi, apa saja yang dilakukan?
6. Hal-hal apa saja yang bisa dilakukan pada saat pengujian sistem?