

BAB II

LANDASAN TEORI

2.1 Helpdesk

Menurut Donna Knapp (2004), definisi *helpdesk* adalah sebuah alat untuk mengatasi persoalan yang didesain dan disesuaikan untuk menyediakan layanan teknis yang dikonsentrasikan untuk produk atau layanan yang spesifik. *Helpdesk* juga dikenal sebagai suatu departemen dalam suatu perusahaan yang digunakan untuk menjawab dan memberikan informasi kepada *user*.

Helpdesk didesain dan disesuaikan untuk *internal support system* dan digunakan untuk mendukung *customer*. Biasanya perusahaan menyediakan layanan *helpdesk* pelanggannya melalui layanan jalur hubungan langsung, situs web, dan email. Ada juga *helpdesk* intern yang menyediakan fasilitas *helpdesk* hanya untuk karyawannya.

Pada umumnya fungsi *helpdesk* mempunyai beberapa fungsi. Fungsi utama adalah menyediakan wadah bagi para pengguna untuk menampung permasalahan dalam berbagai macam komputer. Biasanya *helpdesk* dikelola dengan menggunakan suatu *software*. *Software* ini sering kali menjadi alat yang sangat bermanfaat untuk mencari, menganalisa dan meminimalisasi masalah-masalah tertentu yang umum terjadi pada lingkungan sebuah organisasi (*helpdesk solution*).

Beberapa *helpdesk* mempunyai tingkatan yang berbeda dalam menangani berbagai jenis permasalahan. Pada tingkat pertama *helpdesk* adalah mempersiapkan untuk menjawab pertanyaan yang paling sering dipertanyakan oleh pengguna dan menyediakan solusi berdasarkan dengan dasar pengetahuan.

Helpdesk dalam skala besar umumnya memiliki sebuah *team* yang bertanggung jawab dalam mengatur sistem permasalahan yang berbeda-beda. Beberapa *helpdesk* didalamnya terdapat perhitungan waktu bagi analisis untuk mengerjakan tugas seperti mengkaji masalah dan menentukan penanganan masalah.

IT Helpdesk merupakan sistem manajemen yang digunakan untuk membantu departemen TI untuk menangani kebutuhan dukungan TI bagi perusahaan (Purwanto, 2011). *Customer* dari *IT Helpdesk* ini adalah karyawan-karyawan yang mempunyai permasalahan yang berkaitan dengan teknologi informasi dan biasanya penggunaanya adalah divisi atau departemen yang menangani teknologi informasi pada perusahaan atau organisasi.

2.2 Analisis Sistem (Whitten & Bentley, 2008)

Didefinisikan secara resmi, analisis sistem merupakan kegiatan penelitian atau pembelajaran mengenai suatu sistem dan komponennya. Analisis sistem juga merupakan prasyarat untuk melakukan desain sistem, spesifikasi dari sistem yang baru dan lebih baik. Berdasarkan definisi klasiknya, dapat dilihat bahwa analisis sistem merupakan istilah yang secara kolektif menggambarkan fase awal dari pengembangan sistem. Analisis sistem dibangun oleh pertimbangan bisnis dari pemilik sistem dan pengguna sistem. Sehingga analisis sistem dapat disimpulkan sebagai suatu teknik pemecahan masalah yang menguraikan suatu sistem menjadi bagian-bagian untuk tujuan dalam mempelajari seberapa baik bagian bagian komponen bekerja dan berinteraksi untuk mencapai tujuan sistem yang sebenarnya. Sedangkan analisis sistem informasi adalah tahap pengembangan dalam membangun *project* sistem informasi yang berfokus pada masalah bisnis

dan kebutuhannya, terlepas dari teknologi yang dapat atau yang akan digunakan untuk solusi masalah tersebut.

Analisis sistem mempunyai tahap-tahap yang harus dilakukan. Diantaranya *The Scope Definition Phase*, *The Problem Analysis Phase*, *The Requirements Analysis Phase*, *The Logical Design Phase*, dan *The Decision Analysis Phase*.

2.2.1 *The Scope Definition Phase*

Tahap pendefinisian ruang lingkup merupakan tahap awal dalam proses membangun suatu sistem. Dalam metodologi lain ini biasa disebut dengan tahap penyelidikan awal, tahap studi awal, tahap *survey*, atau tahap perencanaan. Tahap ini dapat menjawab pertanyaan “apakah sistem tersebut berguna untuk dikerjakan?”. Untuk menjawab pertanyaan tersebut, harus mendefinisikan terlebih dahulu ruang lingkup sistem dan permasalahan yang ada, peluang, dan arahan yang memicu sistem tersebut dibangun. Selain meyakinkan bahwa sistem itu layak, pada tahap ini juga harus membangun perencanaan dalam hal skala, strategi pengembangan, menjadwalkan sumber daya dan anggaran.

2.2.2 *The Problem Analysis Phase*

Tahap analisis masalah membantu sistem analis dengan memberikan pemahaman menyeluruh dari masalah, peluang, dan arahan yang memicu sistem tersebut dibangun. Dalam metodologi lain tahap ini biasa dikenal tahap studi, pemahaman sistem yang ada saat ini, tahap investigasi secara mendetail, atau tahap analisis kemungkinan yang bisa digunakan. Tujuan dari tahap ini adalah

mempelajari dan memahami area permasalahan dengan baik untuk dapat menganalisa masalah, peluang dan kendala.

2.2.3 The Requirement Analysis Phase

Tahap analisis kebutuhan merupakan tahap yang mendefinisikan kebutuhan bisnis untuk sebuah sistem baru. Tahap ini dapat menjelaskan tentang apa yang sebenarnya pengguna butuhkan dan yang diinginkan dari sistem baru yang akan dibangun sehingga dapat dikatakan bahwa tahap ini merupakan tahap kritis dalam kesuksesan suatu sistem informasi yang baru.

2.2.4 The Logical Design Phase

Sebuah desain logis dilanjutkan dengan suatu dokumen kebutuhan bisnis dengan menggunakan model sistem yang dapat menggambarkan struktur data, proses bisnis, aliran data, dan tampilan antarmuka.

2.2.5 The Decision Analysis Phase

Tahap analisis keputusan merupakan tahap yang bertujuan untuk mengidentifikasi beberapa solusi pilihan, menganalisis pilihan solusi tersebut, dan merekomendasikan sistem target yang akan dirancang, dibangun, dan diimplementasikan. Kemungkinan-kemungkinan adanya solusi alternatif yang lebih baik selalu ada dalam proses pembangunan sistem. Sangat penting untuk mendefinisikan pilihan, menganalisa dan mencari solusi yang terbaik akan diambil.

2.3 Desain Sistem (Whitten & Bentley, 2008)

Desain sistem adalah spesifikasi dari sebuah solusi detail yang berbasis komputer. Desain sistem informasi didefinisikan sebagai tugas yang berfokus

pada spesifikasi dari solusi detail yang berbasis komputer. Hal tersebut dapat disebut *physical design*. Dengan demikian, dapat dikatakan analisis sistem menekankan masalah bisnis, sedangkan desain sistem berfokus pada masalah teknis atau implementasi sistem. Desain sistem dibangun oleh pertimbangan teknis dari pendesain sistem (System Designer). Karena itu, desain sistem dibangun berdasarkan perspektif dari pendesain sistem. Suatu analisis sistem berfungsi sebagai fasilitator dari desain sistem.

Dari kesimpulan yang diperoleh pada tahap keputusan memicu desain sistem, terdapat dua tujuan tahap desain yang utama. Pertama, analisis berusaha untuk merancang suatu sistem yang memenuhi syarat kebutuhan dan dapat dengan mudah dimengerti oleh *end users*. Kedua, analisis berusaha untuk menyajikan spesifikasi yang jelas dan lengkap untuk para *programmer* dan teknisi. Pada desain sistem ini terdapat 4 tahapan yaitu desain arsitektur sistem, desain database sistem, desain *interface*, dan desain spesifikasi.

2.3.1 Desain Arsitektur Sistem

Tahap awal dari desain sistem adalah menentukan arsitektur aplikasi atau sistem. Arsitektur aplikasi mendefinisikan teknologi yang akan digunakan oleh satu, lebih, atau semua sistem informasi dalam hal data, proses, antarmuka, dan komponen jaringan.

2.3.2 Desain Database

Tahap selanjutnya dalam mendesain sistem adalah merancang spesifikasi database yang sesuai. Database merupakan sebuah sumber daya bersama. Beberapa aplikasi memungkinkan untuk menggunakan database yang sama.

Perancang atau pendesain sistem juga harus menganalisa bagaimana aplikasi dapat mengakses data untuk meningkatkan performa.

2.3.3 Desain Antarmuka

Setelah proses desain database, desainer dapat bekerja sama dengan pengguna sistem untuk mengembangkan desain input, output dan spesifikasi dialog. Desain antarmuka penting dilakukan untuk menunjang pengguna dan manajer untuk dapat menggunakan aplikasi dengan mudah tanpa adanya kebingungan dalam melakukan input maupun membaca output.

2.3.4 Desain Spesifikasi

Tahap terakhir dalam desain sistem ini melibatkan keseluruhan spesifikasi dari tahap sebelumnya menjadi satu set spesifikasi yang akan memandu *programmer* selama tahap pengembangan metodologi sistem. Tahap ini menjelaskan infrastruktur yang akan digunakan pada saat sistem diimplementasikan.

2.4 Mean Time To Repair (MTTR)

Menurut Torrel & Avelar (2010), MTTR, atau *Mean Time To Repair* adalah waktu yang diperlukan untuk memulihkan suatu sistem dari sebuah kegagalan. Dalam hal ini juga termasuk waktu yang dibutuhkan dalam mendiagnosa masalah, waktu yang dibutuhkan untuk mendapatkan teknisi, dan waktu yang diperlukan untuk memperbaiki sistem (*hardware*). Sama dengan MTBF, MTTR diwakili dalam satuan jam. MTTR menunjukkan nilai *availability* dan bukan *reliability* seperti MTBF. Semakin lama MTTR atau semakin tinggi nilai MTTR maka semakin buruk. Sederhananya, jika dibutuhkan waktu lebih

lama untuk memulihkan sebuah sistem dari kegagalan atau kerusakan, maka sistem ini memiliki ketersediaan (*availability*) yang lebih rendah.

MTTR adalah parameter yang berguna yang harus digunakan di awal perencanaan dan perancangan tahap dari suatu sistem. Parameter yang digunakan dalam menilai aksesibilitas / lokasi komponen sistem, misalnya sebuah komponen yang sering gagal harus ditempatkan di mana ia dapat dengan mudah dihapus dan diganti.

MTTR juga dapat memberikan informasi untuk penentuan perangkat teknologi mana yang harus diberikan cadangan dan mana yang tidak. Nilai MTTR yang sangat tinggi dapat digunakan sebagai acuan bahwa perangkat teknologi tersebut sebaiknya diberikan cadangan. Pemberian cadangan ini bertujuan agar operasional perusahaan dapat segera berjalan normal ketika terjadi kerusakan. Untuk menghitung perkiraan dari nilai MTTR ini adalah sebagai berikut :

$$MTTR = \frac{\sum t}{n} \dots\dots\dots(2.1)$$

MTTR = Mean Time To Repair

t = Waktu yang diperlukan untuk reparasi

n = Jumlah reparasi yang pernah dilakukan.

Berikut contoh perhitungan untuk mencari nilai MTTR suatu perangkat atau mesin dengan data *history* dalam 7 hari terakhir:

Tabel 2.1. Tabel Data *history* mesin/perangkat (MTTR)

Hari	Waktu Kerja Mesin (Jam)	Masa Perbaikan (Jam)
1	8	0
2	8	2

3	8	0
4	8	3
5	8	0
6	8	0
7	8	2

Dari tabel 2.1 dapat diketahui jumlah kerusakan yang terjadi sebanyak 3 kali (dalam 7 hari) dan lama waktu perbaikan adalah $(2+3+2)= 7$.

$$MTTR = \frac{(2 + 3 + 2)}{3} = \frac{7}{3} = 3,5$$

Jadi nilai MTTR (waktu rata-rata yang diperlukan untuk memperbaiki sistem) adalah 3,5 jam.

2.5 Mean Time Between Failure (MTBF)

MTBF, atau *Mean Time Between Failure* (Torrel & Avelar, 2010) adalah ukuran dasar dari keandalan sistem. MTBF merupakan waktu rata-rata yang dibutuhkan oleh sistem untuk bekerja tanpa mengalami kegagalan dalam periode tertentu. Perkiraan nilai MTBF juga dapat memberikan informasi mengenai keandalan suatu perangkat TIK dimana dalam permasalahan ini juga dapat dianalisa tentang kemungkinan *human error* yang mengakibatkan kegagalan perangkat TIK.

MTBF biasanya direpresentasikan dalam satuan jam. Semakin tinggi jumlah MTBF, semakin tinggi keandalan suatu sistem atau produk. Bagi produsen, nilai MTBF ini sangat penting dalam proses pengambilan keputusan, karena dari nilai MTBF maka dapat diketahui masa hidup suatu produk. Pengambilan keputusan ini menyangkut pemilihan produk yang nantinya akan digunakan untuk mendukung suatu sistem yang ada.

Nilai MTBF dapat dihitung atau diukur dengan membagi antara total waktu masa optimal dengan jumlah kerusakan yang terjadi. Berikut persamaan untuk menghitung nilai MTBF :

$$MTBF = \frac{\sum tUptime}{n} \dots\dots\dots(2.2)$$

MTBF = Mean Time Between Failure

tUptime = Waktu optimal

n = Jumlah kerusakan yang terjadi

Berikut contoh perhitungan untuk mencari nilai MTBF suatu perangkat atau mesin dengan data *history* dalam 7 hari terakhir:

Tabel 2.2. Tabel Data *history* mesin/perangkat (MTBF)

Hari	Waktu Kerja Mesin (Jam)	Masa Perbaikan (Jam)	Masa Optimal (Jam)
1	8	0	(8-0) = 8
2	8	2	6
3	8	0	8
4	8	3	5
5	8	0	8
6	8	0	8
7	8	2	6

Dari tabel 2.2 dapat diketahui jumlah kerusakan yang terjadi sebanyak 3 kali (dalam 7 hari) dan masa optimal perangkat adalah (8+6+8+5+8+8+6)= 49.

$$MTBF = \frac{(8 + 6 + 8 + 5 + 8 + 8 + 6)}{3} = \frac{49}{3} = 16\frac{1}{3}$$

Jadi nilai MTBF (waktu rata-rata yang dibutuhkan oleh sistem untuk bekerja tanpa mengalami kegagalan dalam periode tertentu) adalah $16\frac{1}{3}$ jam.

2.6 Keandalan (*Reliability*)

Dasar pemikiran konsep analisa keandalan adalah bertolak dari pemikiran layak atau tidaknya suatu sistem melakukan fungsinya. Keandalan / *Reliability* (Torrel & Avelar, 2010) dapat didefinisikan sebagai nilai probabilitas bahwa suatu komponen atau sistem akan sukses menjalani fungsinya, dalam jangka waktu dan kondisi operasi tertentu.

Keandalan digunakan sebagai indikator dari tingkat layanan suatu produk atau jasa. Keandalan suatu layanan dianggap sempurna jika setiap kali produk atau jasa tersebut digunakan tanpa ada kegagalan yang terjadi. Dengan mengetahui informasi tentang keandalan suatu sistem atau produk diharapkan dapat memberikan indikator keandalan suatu layanan produk atau jasa sehingga dapat dilakukan evaluasi tingkat atau kualitas layanannya.

MTBF merupakan ukuran dasar dari keandalan sistem. MTBF ini biasanya direpresentasikan dalam satuan jam. Jika dapat diketahui nilai dari MTBF maka dapat diketahui pula nilai *reliability*. Nilai MTBF ini digunakan untuk menentukan nilai *failure rate* yang digunakan pada perhitungan nilai *reliability*.

$$\lambda = \frac{1}{MTBF} \dots\dots\dots(2.3)$$

λ = Tingkat Kegagalan (*failure rate*)

MTBF = Mean time between failure

Berikut adalah persamaan dari *Reliability* :

$$R = e^{-(\lambda t)} \dots\dots\dots(2.4)$$

$$R = e^{-\left(\frac{1}{MTBF} t\right)} \dots\dots\dots(2.5)$$

Berdasarkan data pada tabel 2.2, dapat diketahui nilai dari MTBF. Dengan mengetahui nilai MTBF, maka dapat dihitung nilai *reliability* pada periode (t) tertentu sebagai berikut :

Jika :

MTBF = 16,33 jam

$t = 12$ jam, maka *reliability* = 0,47 :

$$R = e^{-\frac{1}{16,33} 12} = 0,47$$

Jadi dapat disimpulkan bahwa pada $t = 12$ jam nilai *reliability* adalah 0,47. Hal ini berarti dalam kurun waktu 12 jam perangkat tersebut mempunyai kemungkinan dapat bekerja dengan baik dengan nilai probabilitas sebesar 0,47. Nilai dari *reliability* merupakan nilai probabilitas antara 0 sampai dengan 1, dengan pengertian nilai 0 merupakan kondisi dimana tidak dapat terjadi dan 1 merupakan kondisi yang pasti terjadi. Nilai probabilitas *reliability* 0,47 merupakan nilai yang mendekati 0,50 yang berarti kemungkinan untuk terjadi dan tidak terjadi hampir sama, dalam artian peluang perangkat tersebut dapat bekerja dengan baik sesuai dengan fungsinya adalah 0,47.

2.7 Ketersediaan (*Availability*)

Definisi ketersediaan (Torrel & Avelar, 2010) adalah probabilitas bahwa perangkat akan melakukan fungsi yang diperlukan tanpa kegagalan dalam kondisi persyaratan untuk jangka waktu tertentu. Sebelum ketersediaan sistem dapat ditentukan, ketersediaan perangkat yang harus dipahami. Penting untuk diingat bahwa setiap perangkat akan memiliki probabilitas kegagalan.

Ada dua faktor utama yang terlibat dalam perhitungan ketersediaan: *Mean Time Between Failure* (MTBF) dan *Mean Time To Repair* (MTTR). MTBF diperoleh dari membagi antara total waktu masa optimal dengan jumlah kerusakan yang terjadi. MTTR adalah waktu rata-rata untuk memperbaiki dan mengembalikan perangkat untuk kembali ke keadaan normal.

Setelah MTBF dan MTTR diketahui, ketersediaan komponen dapat dihitung dengan menggunakan rumus berikut:

$$\text{Availability} = \frac{\text{MTBF}}{\text{MTBF} + \text{MTTR}} \dots\dots\dots(2.6)$$

Nilai dari *availability* dan *reliability* sangat penting untuk dituntut setinggi mungkin bahkan kalau bisa dapat mencapai nilai sempurna. Suatu layanan yang baik tentu dapat memberikan nilai lebih bagi suatu perusahaan. Layanan dalam hal ini dapat berupa kinerja suatu sistem atau kinerja dari manusia. Kinerja yang baik dari suatu sistem dapat memberikan kelancaran operasional bagi perusahaan dimana kelancaran tersebut dapat memberikan keuntungan, baik berupa materi maupun kemudahan dalam proses bisnis.

Berdasarkan data tabel 2.1 dan tabel 2.2, dapat diketahui nilai MTTR dan MTBF, sehingga dapat diperoleh nilai dari *availability* sebagai berikut :

$$A = \frac{16,33}{(16,33 + 3,5)} = 0,82$$

Jadi nilai *availability* yang diperoleh adalah 0,82. Hal ini berarti nilai probabilitas suatu sistem atau perangkat dengan data pada tabel 2.1 dan 2.2 adalah 0,82. *Availability* sama dengan nilai *reliability* yaitu berada pada nilai antara 0 sampai dengan 1, dengan pengertian nilai 0 merupakan kondisi dimana tidak dapat terjadi dan 1 merupakan kondisi yang pasti terjadi. Nilai probabilitas 0,82

merupakan nilai yang mendekati 1 yang berarti kemungkinan untuk terjadi lebih tinggi dari pada tidak terjadi, dalam artian peluang perangkat tersebut dapat bekerja ketika dibutuhkan adalah 0,47.

2.8 Kegagalan Teknologi Informasi

Untuk melakukan peninjauan mengenai nilai MTBF dan MTTR maka diperlukan definisi dari kegagalan yang terjadi. Kegagalan disini berarti terjadi kerusakan atau permasalahan dalam suatu sistem atau perangkat teknologi. Kegagalan (Torrel & Avelar, 2010) mempunyai dua definisi dasar:

1. Berakhirnya kemampuan produk secara keseluruhan untuk melakukan fungsinya.
2. Berakhirnya kemampuan setiap komponen untuk melakukan fungsi yang diperlukan tetapi dapat menyebabkan berhentinya fungsi suatu produk atau sistem.

Jadi dapat disimpulkan bahwa kegagalan yang digunakan adalah kegagalan yang dapat menyebabkan sistem terganggu. Kerusakan komponen yang tidak menyebabkan sistem terganggu atau tidak dapat digunakan tidak termasuk kegagalan yang didefinisikan untuk menghitung nilai MTTR dan MTBF.

2.9 Penentuan Prioritas Masalah

Masalah adalah kesenjangan antara apa yang diharapkan (*expected*) dengan apa yang aktual terjadi (*observed*). Idealnya, semua permasalahan yang timbul harus dicarikan jalan keluarnya. Namun, karena keterbatasan sumber daya, dana, dan waktu menyebabkan tidak semua permasalahan dapat dipecahkan sekaligus, untuk itu perlu ditentukan masalah yang menjadi prioritas. Setelah pada

tahap awal merumuskan masalah, maka dilanjutkan dengan menetapkan prioritas masalah yang harus dipecahkan. Prioritas masalah didapatkan dari data atau fakta yang ada secara kualitatif, kuantitatif, subjektif, objektif serta adanya pengetahuan yang cukup.

Menurut Azwar (1996), penetapan prioritas dinilai oleh sebagian besar manager sebagai inti proses perencanaan. Langkah yang mengarah pada titik ini, dapat dikatakan sebagai suatu persiapan untuk keputusan penting dalam penetapan prioritas. Sekali prioritas ditetapkan, langkah berikutnya dapat dikatakan merupakan gerakan progresif menuju pelaksanaan. Dalam penentuan prioritas, aspek penilaian dan kebijaksanaan banyak diperlukan bersama-sama dengan kecakapan unik untuk mensintesis berbagai rincian yang relevan.

2.10 Metode Delbec

Menurut Intiasari (2011), metode ini memprioritaskan masalah yang dilakukan dengan memberikan bobot (yang merupakan nilai maksimum dan berkisar antara 0 sampai 100) dengan kriteria:

1. Besar masalah yaitu % atau jumlah atau kelompok penduduk yang ada kemungkinan terkena masalah serta keterlibatan masyarakat dan instansi terkait.
2. Kegawatan masalah yaitu tingginya angka morbiditas dan mortalitas, kecenderungannya dari waktu ke waktu.
3. Biaya / dana yaitu besar atau jumlah dana yang diperlukan untuk mengatasi masalah baik dari segi instansi yang bertanggung jawab terhadap penyelesaian masalah atau dari masyarakat yang terkena masalah.

4. Kemudahan yaitu tersediannya tenaga, sarana / peralatan, waktu serta cara atau metode dan teknologi penyelesaian masalah seperti tersediannya kebijakan / peraturan, petunjuk pelaksanaan (juklak), petunjuk teknis (juknis) dan sebagainya.

Langkah-langkah yang harus dilakukan sebagai berikut:

1. Tentukan dahulu bobot masing-masing kriteria.
2. Menentukan skala atau skor dari variabel yang akan ditentukan nilai prioritasnya.
3. Isi setiap kolom dengan hasil perkalian antara bobot dengan skor masing-masing masalah. Besarnya skor tidak boleh melebihi bobot yang telah disepakati. Bila ada perbedaan pendapat dalam menentukan besarnya bobot dan skor yang dipilih rata-ratanya. Kemudian, jumlahkan nilai masing-masing kolom tersebut sehingga menjadi sebuah nilai prioritas.

