

Bab 2

Landasan Teori

2.1 Sistem Informasi

Sistem informasi merupakan sebuah susunan yang meliputi data, proses, orang serta teknologi informasi yang terhubung dan bertujuan untuk menyatukan, merubah, menyimpan dan menyediakannya sebagai *output* informasi yang diperlukan untuk mendukung bisnis sebuah organisasi. Sistem informasi dapat digunakan merekam dan mengelola data agar menghasilkan informasi pendukung sebuah organisasi. Selain itu sistem informasi juga merupakan kombinasi terorganisir yang meliputi orang, perangkat lunak dan keras, sumber data, kebijakan dan prosedur[6]. Banyak organisasi yang menganggap perlu adanya sistem informasi untuk memiliki kemampuan bersaing dan keuntungan bersaing.

2.2 Stakeholder Sistem

Stakeholder merupakan orang yang mempunyai ketertarikan pada sistem informasi yang ada atau sudah ditawarkan sebelumnya. Beberapa orang yang dapat dianggap sebagai *stakeholder* pekerja yang melakukan hal teknis maupun nonteknis ataupun bisa juga pekerja di dalam maupun luar sistem[7]. *Stakeholder* dalam sistem informasi bisa dibagi menjadi lima kelompok sebagai berikut:

1. Pemilik sistem

Biasanya pemilik sistem ini memiliki tanggung jawab untuk proses pendanaan proyek pengembangan, pengoprasian dan perawatan pada sistem informasi yang bermula dari tingkat manajemen. Pemilik sistem biasanya tertarik pada berapa biaya pembuatan sistem dan keuntungan yang bisa didapat.

2. Pengguna sistem

Pengguna sistem adalah mayoritas pekerja yang nantinya akan menggunakan atau terpengaruh oleh sistem yang bertugas untuk menangkap, memvalidasikan, memasukan, merespon, menyimpan dan bertukar data. Berbeda dengan pemilik sistem, pengguna sistem akan lebih melihat sistem yang tersedia untuk memudahkan pekerjaan mereka serta memudahkan pembelajaran penggunaan sistem.

3. Desainer sistem

Desainer sistem adalah spesialis teknologi sistem informasi, mereka lebih tertarik pada pilihan teknologi informasi dan desain sistem yang menggunakan teknologi terpilih. Beberapa dari mereka cenderung lebih fokus pada keahlian khusus yang bersikap teknis.

4. Pembangun sistem

Mereka merupakan spesialis teknologi sistem informasi yang bertugas untuk membuat sistem dan komponen-komponen yang ada didalamnya yang dilakukan berdasarkan detail desain yang dibuat oleh desainer sistem.

5. Analis sistem

Peran analis sistem adalah melengkapi peran dari semua *stakholder* lainnya, mereka merupakan orang yang mempelajari tentang kebutuhan suatu organisasi agar dapat mengelola manusia, proses, data dan teknologi informasi yang ada demi kemajuan bisnis dengan baik.

2.3 Pengembangan Sistem Informasi

Pengembangan sistem merupakan suatu proses dimana sekumpulan metode, aktivitas dan peralatan terotomasi yang diperlukan oleh *stakeholder* agar dapat mengembangkan sistem secara lanjut serta dapat memperbaiki sistem informasi maupun perangkat lunak yang ada. Pengembangan sistem informasi memerlukan

kerangka untuk bisa sistem tersebut dikembangkan, *Capability Maturity Model* (CMM) merupakan kerangka yang biasa digunakan untuk menilai tingkat kecukupan pengembangan sistem yang ada pada organisasi sebuah perusahaan baik itu dari proses-proses manajemen maupun dari proses produksi. Terdapat 5 tingkatan kematangan dalam CMM, berikut adalah tingkatan tersebut[7]:

1. Tingkat Permulaan

Pada tahap ini pengembangan sistem tidak melakukan proses secara berurutan dan konsisten. Tim yang melakukan proses pengembangan hanya menggunakan peralatan dan metode yang mereka ketahui. Hasil dari pengembangan ini tergantung dari seberapa terampil dan berpengalamannya tim tersebut., proses ini tidak dapat ditebak dan tidak dapat diulang.

2. Tingkat Pengulangan

Pada tahap ini pengembangan selalu memiliki variasi yang di dapat dari proyek-proyek yang sudah dilakukan. Praktik-praktik manajemen proyek yang efektif meletakkan fondasi untuk proses-proses terstandarisasi pada tingkat berikutnya.

3. Tingkat Terdefiniskan

Pada tahap ini, sistem standar dikembangkan dan semua proyek menggunakan versi disesuaikan. Hal ini dilakukan untuk mengembangkan dan merawat sistem informasi yang ada pada perangkat lunak.

4. Tingkat Terkelola

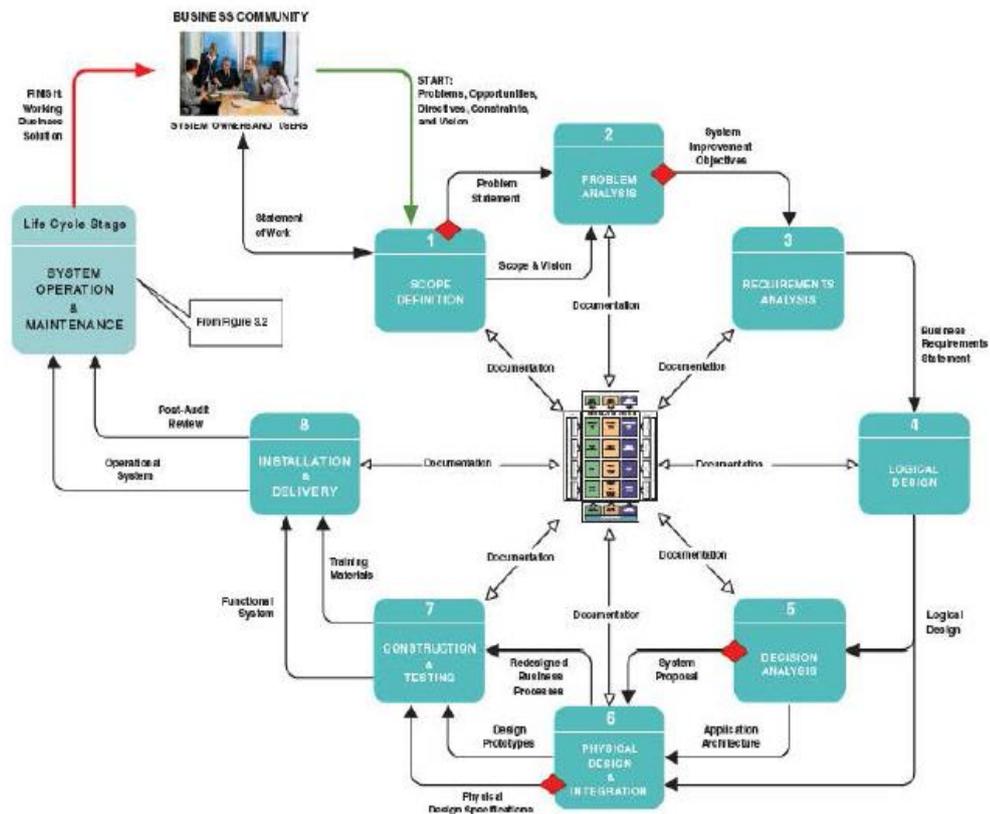
Pada tahap ini, tujuan-tujuan dari pengembangan sistem dapat diukur untuk kualitas dan produktivitas yang akan dibentuk. Pengembangan sistem standar dan kualitas produk diukur secara terinci dan rutin berdasarkan data terkumpul. Manajemen lebih pro aktif daripada reaktif terhadap masalah-masalah pengembangan sistem.

5. Tingkat Teroptimisasi

Pada tahap ini pengembangan sistem distandarisasi secara berkelanjutan dan terus diperbaiki berdasarkan ukuran dan analisis data yang sudah dibentuk pada tingkat 4.

2.4 Fase Proyek Klasik

Metode yang banyak digunakan dalam pengembangan sistem adalah siklus hidup pengembangan perangkat lunak (SDLC) yang merupakan metode yang digunakan perangkat lunak berkualitas untuk dikembangkan dalam jangka waktu tertentu dan sesuai dengan keinginan pelanggan[8]. Berbeda dengan metode SDLC terdapat metodologi pengembangan sistem yang lain yaitu metodologi dengan menggunakan kerangka FAST (*Framework for the Application of Systems Technique*). Banyaknya referensi dan metodologi yang ditemui dari proses gabungan praktek-praktek maka kerangka FAST ini dikembangkan. Hipotesis yang ada pada metodologi ini tidak menggunakan pendekatan tunggal pada proses analisis sistem yang dilakukan, akan tetapi metodologi ini akan menyatukan semua pendekatan yang sudah ada (*agile methode*). Setiap fase yang ada pada setiap metodologi sangat bervariasi, setiap fase klasik dari metodologi FAST menghasilkan *output* ke fase berikutnya. Berikut ini adalah ilustrasi fase klasik dari metodologi FAST dapat dilihat pada gambar 2.1.



Gambar 2.1 Pandangan proses akan pengembangan sistem

1. *Scope Definition* (Definisi Lingkup)

Pada proses perancangan sistem informasi fase ini merupakan hal yang harus dilakukan pertama kali. Pada fase ini sistem yang ada dirancang dan dibuat akan didefinisikan ruang lingkungnya. Fase definisi lingkup ini merupakan tahap pertama dalam proses pengembangan, pada pembahasan yang lain fase ini sering disebut juga dengan fase penyelidikan pendahuluan. Pada fase ini informasi akan dikumpulkan yang nantinya akan menentukan ruang lingkup proyek yang ada pada sistem. Proses ini dikerjakan untuk menemukan pokok masalah ada pada proyek, selain itu proses ini juga dapat memberikan kesempatan organisasi untuk bisa meningkatkan kinerja dan mengetahui kebutuhan-kebutuhan baru yang diberikan oleh pihak manajemen.

Langkah awal yang harus dikerjakan dalam fase definisi lingkup adalah dengan mengidentifikasi masalah-masalah, kesempatan-kesempatan, perintah dan batasan. Namun hal tersebut harus berdasarkan urgensi, visibilitas, manfaat-manfaat yang nyata dan prioritas. Lingkup juga mendefinisikan batasan-batasan proyek atau aspek-aspek bisnis[7].

2. *Problem Analysis* (Analisis Permasalahan)

Pada tahap ini masalah yang telah ditemukan didalam sistem akan dianalisis dan dicari permasalahannya. Hasil yang diinginkan dari tahap ini performa dari sistem akan meningkat sehingga pihak perusahaan mendapatkan keuntungan. Dengan adanya fase ini tujuan yang ingin dicapai adalah masalah yang ada dapat dipelajari dan dipahami secara menyeluruh sehingga dapat mengetahui batasannya[7].

3. *Requirements Analysis* (Analisis Persyaratan)

Hal yang dilakukan pada tahap ini adalah mengurutkan kebutuhan bisnis dan membuat prioritasnya. Tahapan ini bertujuan untuk menemukan data, proses serta desain tampilan sistem yang diinginkan oleh pengguna. Pada tahap ini juga mulai terlihat adanya berbagai solusi-solusi alternatif, khususnya solusi yang berkaitan dengan masalah teknis. Persyaratan pada sistem terbagi menjadi dua yaitu[7]:

- a. Persyaratan fungsional biasa dibuktikan dalam istilah masukan, keluaran, proses dan data yang diperlukan dalam meningkatkan sistem yang ada.
- b. Persyaratan nonfungsional merupakan penjelasan mengenai kemampuan, batasan dan karakteristik sistem yang dapat menentukan sistem dapat membantu pengguna atau tidak. Pada persyaratan nonfungsional dapat dilakukan pengklasifikasian berdasarkan kerangka PIECES (Performa, Informasi, Ekonomi, Kontrol, Efisiensi dan Pelayanan)

4. *Logical Design* (Desain Logis)

Istilah desain logis harus dipekenalkan sebagai teknologi yang dapat melakukan tugasnya sendiri, Tahapan ini bertujuan untuk merubah kebutuhan bisnis menjadi model sistem yang akan dibuat. Desain logis sendiri dalam tingkatan yang lebih lanjut dapat mengumpulkan dan menyimpan persyaratan bisnis berbentuk model untuk menggambarkan sistem dari proses bisnis, struktur dan aliran datanya[7].

5. *Decision Analysis* (Analisis Keputusan)

Tahap ini dilakukan untuk mengetahui, menganalisis dan memberikan solusi sebuah sistem yang dirancang. Pada tahap ini akan ditentukan perangkat keras dan perangkat lunak yang akan dipakai pada proses implementasi[7].

6. *Physical Design* (Desain Fisik)

Tahap ini bertujuan untuk merubah kebutuhan bisnis menjadi desain logis dan desain fisik yang nantinya desain ini akan digunakan sebagai dasar dalam merancang dan mengembangkan sistem. Dalam desain logis ini solusi sangat bergantung pada hal teknis, sedangkan desain fisik akan memberikan solusi teknis yang lebih spesifik dari desain logis[7]. Pembuatan model data fisik ini juga bertujuan untuk menggambarkan struktur penyimpanan data yang akan dibuat pada *database*.

7. *Construction and Testing*

Setelah proses pemuatan desain fisik, maka langkah selanjutnya yang dilakukan adalah membangun sistem dan melakukan tahap uji coba sesuai dengan kebutuhan bisnis perusahaan. Dalam tahap ini pembuatan basis data dan desain antar muka mulai dibuat. Pengujian ini bertujuan untuk melihat sistem secara keseluruhan sebelum siap diimplementasikan[7].

8. *Installation and Delivery*

Pada tahap ini sistem yang dibuat sudah mulai dioperasikan. Langkah awal yang dilakukan adalah dengan mengembangkan sistem yang telah dibangun. Pengembangan sistem ini dimaksudkan untuk memberikan bantuan kepada pengguna sistem agar terbiasa dengan sistem. Selain itu pada tahap ini perangkat lunak harus memperhatikan proses instalasi databasenya[7].

2.5 Flow Map

Fungsi dari *flow maps* adalah untuk mendefinisikan hubungan antar bagian yang ada pada sebuah proses dalam bentuk aliran data. Penggunaan *flow maps* ini dapat membantu analisis dalam membagi masalah yang ada di dalam sebuah proses menjadi segmen-segmen yang lebih kecil sehingga dapat memberikan alternatif pengoperasian yang dilakukan. *Flow maps* sendiri dalam suatu sistem informasi sering disebut aliran data yang berbentuk dokumen ataupun formulir dalam sebuah aktivitas yang saling berkaitan antar kebutuhan data dan informasinya. Proses aliran dokumen yang terjadi di sistem dapat berupa entitas yang ada didalam maupun diluar sistem. Simbol-simbol *flow map* yang digunakan dapat dilihat pada gambar 2.2.

Symbol	Nama Simbol	Keterangan
	Dokumen	Digunakan untuk menggambarkan semua jenis dokumen yang merupakan formulir yang digunakan untuk mengentry data keluarga.
	Proses	Merupakan kegiatan proses dari operasi program computer
	Proses manual	Merupakan proses manual pada flowmap
	File Harddisk	Merupakan media penyimpanan dari proses entry data dan proses komputerisasi
	Offline Storage	Merupakan tempat penyimpanan data berupa arsip
	Garis Alir	Merupakan arus data
	Keyboard	Merupakan proses penyimpanan menggunakan keyboard

Gambar 2.2 Simbol *flow map*

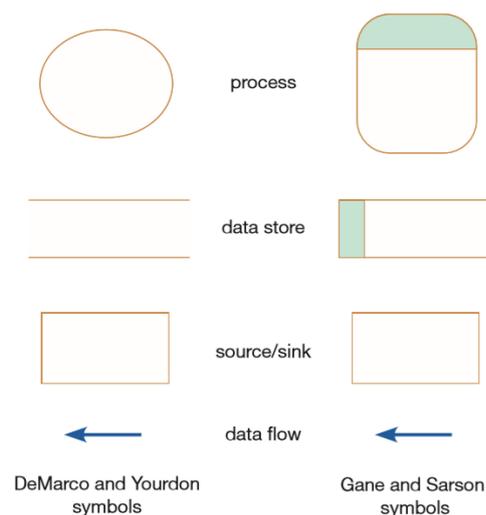
2.6 Data Flow Diagram (DFD)

Diagram ini merupakan diagram yang dapat digunakan dalam berbagai macam kondisi. Dengan hanya empat simbol notasi, DFD dapat mewakili sistem informasi baik dari bentuk fisik maupun logis[9]. *Data Flow Diagram* sendiri memiliki notasi-notasi dalam penggambaran alirannya yg ada pada sistem, notasi ini dimaksudkan untuk membantu memahami sistem secara jelas dan terstruktur. Tujuan dibuatnya *data flow diagram* adalah:

1. Memberikan tanda suatu data apabila terjadi perubahan pada saat data tersebut berpindah melalui sistem.
2. Membuat fungsi dan sub fungsi data yang berubah pada aliran sistem.

Diagram DFD dapat digunakan sebagai alat pembuatan model yang nantinya dapat membantu perancang untuk menggambarkan jaringan sistem yang saling diintegrasikan dengan aliran data data. DFD juga berfungsi sebagai alat yang berfokus pada aliran data yang memiliki konsep dekomposisi, hal ini dapat mempermudah penggambaran sistem sehingga para professional mudah untuk mengkomunikasikan sistem kepada pengguna maupun perancang sistem.

Ada dua set standar simbol DFD yang berbeda, setiap set terdiri dari empat simbol yang mewakili hal-hal yang sama: aliran data, penyimpanan data, proses, dan sumber atau sink (entitas eksternal). Himpunan simbol yang akan biasa digunakan dalam merancang adalah Gane dan Sarson (1979). Perangkat standar lainnya dikembangkan oleh DeMarco (1979) dan Yourdon (Yourdon and Constantine, 1979). Berikut merupakan simbol DFD yang digunakan dan dapat dilihat pada gambar 2.3 dibawah ini.



Gambar 2.3 Simbol DFD

Arti dari simbol *data flow diagram* adalah sebagai berikut:

1. *Process*

Process adalah tanggapan dari masuknya aliran data. Notasi ini disimbolkan dengan bentuk persegi yang bersudut bulat.

2. *Data store*

Data store digunakan untuk menyimpan data yang ditangkap dari sistem informasi yang ada. Notasi dari *data store* sendiri dinyatakan dengan kotak *open-end*.

3. *Source* atau *link*

Notasi ini dapat mendefinisikan organisasi, unit, orang, atau sistem lain yang berada diluar lingkup sistem ini tetapi masih berhubungan dalam menerima *input* dan *output* dari sistem.

4. *Data flow*

Notasi ini menunjukkan masuknya data ke dalam proses maupun berupa hasil dari data yang sudah diproses.

2.7 Diagram Dekomposisi

Diagram ini digunakan untuk menguraikan suatu sistem kedalam bagian yang lebih spesifik dan detail, diagram ini juga biasa disebut dengan grafik hierarki. Dekomposisi diagram juga dapat berfungsi sebagai alat perencanaan untuk memodelkan proses yang disebut dengan *data flow diagram*. Diagram dekomposisi dan aliran data diagram adalah alat yang efektif untuk mengidentifikasi proses. Tapi tidak menunjukkan logika dalam proses-proses tersebut. Diagram ini juga efektif digunakan untuk berkomunikasi dengan pengguna maupun programmer sistem[7]. Dalam diagram dekomposisi terdapat penurunan level yang memiliki arti jika semakin rendah maka diagram menjelaskan proses secara spesifikasi dan jelas.

2.8 Permodelan *Use Case*

Use case merupakan model yang biasa digunakan dalam mengembangkan suatu perangkat lunak untuk mengetahui kebutuhan fungsional sistem, *Use case* digunakan untuk menjelaskan secara naratif bagaimana interaksi antara sistem dan aktor yang terlibat[10]. *Use case* bertujuan untuk mendeskripsikan atau memodelkan proses yang saling berkaitan dan entitas rangkaian skenario gabungan[11]. Pendeskripsian tersebut meliputi sistem, lingkungan, maupun hubungan antara sistem dengan lingkungan yang ada disekitarnya. *Use case* memiliki beberapa komponen yang dapat dilihat sebagai berikut:

1. *Actor*

Actor digambarkan sebagai seseorang yang berhubungan dengan sistem, aktor ini memiliki peran untuk menerima dan memasukan informasi dari sistem. Simbol aktor biasanya digambarkan dengan notasi *stickman*. Berikut merupakan simbol aktor yang dapat dilihat pada gambar 2.4.



Gambar 2.4 Simbol aktor

2. *Use case*

Use case merupakan gambaran aktivitas yang ada pada sistem, *use case* ini digunakan agar pengguna dapat lebih mengerti penggunaan sistem.



Gambar 2.5 Simbol *use case*

2.9 Microsoft Access

Microsoft Access adalah sistem yang biasa digunakan untuk menampilkan dan mengelola *database*. Pengguna dapat mengelola berbagai macam jenis *database* dengan mudah tetapi dengan kapasitas penyimpanan data yang tidak besar. Microsoft Access sendiri telah dilengkapi dengan kemampuan untuk membuat laporan dan menyediakan kemudahan dalam pengembangan aplikasi karena hanya memerlukan perangkat yang sederhana[12]. Perangkat lunak ini merupakan anggota dari aplikasi Microsoft. Microsoft Access dapat mengolah berbagai macam format data seperti Microsoft AQL Server, Oracle Database atau Microsoft Jet Database Engine.

2.10 *Point of Sale* (POS)

Point of sale telah menjadi menjadi populer dan permintaan perangkat lunak POS meningkat dari hari ke hari di arena bisnis karna menawarkan metode yang cocok untuk kebutuhan transaksi pengguna. Saat ini, perangkat lunak *Point of Sale* banyak digunakan secara luas dalam bisnis ritel. Perangkat lunak ini telah mengubah sistem bisnis yang masih manual ke sistem terkomputerisasi. Tidak dapat dipungkiri bahwa sistem manual masih memiliki banyak kelemahan seperti memakan waktu yang lama, kurangnya keamanan dan lainnya. Akibatnya sering terjadi kesalahan

dalam proses pendataan. Jadi, perangkat lunak POS dikembangkan untuk menyelesaikan kesulitan mereka[13].

Perangkat lunak *Point of Sale* biasanya merupakan sistem komputerisasi yang mengelola penjualan dan melakukan operasi sederhana di toko ritel[14]. Sistem pada perangkat lunak *Point of Sale* sendiri biasanya merupakan perangkat komputer yang dapat dihubungkan dengan pemindai barcode dan perangkat cetak seperti printer, di mana pada komputer tersebut telah terinstal perangkat lunak POS. beberapa contoh penggunaan perangkat POS adalah seperti transaksi kasir atau pada proses pembayaran di minimarket, hotel restoran dan sebagainya. Sistem POS ini dapat berdiri sendiri yaitu tidak terhubung dengan sistem POS lainnya atau dapat juga dirancang untuk saling terhubung melalui jaringan lokal maupun internet jika diperlukan[15]. Terdapat juga POS yang sudah menggunakan metode *cloud*, tetapi pada proses implementasinya banyak pengguna yang masih ketakutan dengan masalah keamanan dan privasi dari data yang belum memungkinkan *cloud* untuk digunakan secara maksimal[16].