BAB II

TINJAUAN PUSTAKA

2.1 Profil Tempat Kerja Praktek

2.1.1 Sejarah Instansi

Pada tahun 2005 PT. SMT Solution Indonesia awalnya bergerak dalam bidang Trading, dan proses trading berlangsung selama 2 tahun mulai 2005 sampai 2007. Kemudian pada tahun 2007 PT. SMT Solution Indonesia sudah tidak bergerak dalam bidang Trading lagi, melainkan begerak dalam bidang SMT (Surface Mount Technology) yaitu teknologi dalam pemasangan komponen-komponen pada PCB.

PT. SMT Solution Indonesia menjalin kerja sama dengan PT. SAMSUNG yang berpusat di Korea. Pada awalnya PT. SMT Solution Indonesia tidak langsung menjalin kerja sama dengan PT. SAMSUNG pusat, tetapi PT. SMT Solution Indonesia menjalin kerja sama dengan Value Plus dimana Value Plus merupakan perusahaan yang sebagian besar mendesain produk-produk PT. SAMSUNG, tetapi rancangan suatu produk yang akan dibuat itu langsung dari pihak PT. SAMSUNG, dan Value Plus hanya bertugas dalam mendesain produk sesuai dengan permintaan PT. SAMSUNG.

Sesuai dengan bertambah banyaknya permintaan untuk suatu produk yang didesain, Dengan demikian Value Plus menjalin kerja sama dengan PT. SMT Solution Indonesia dengan tujuan untuk menangani proses SMT itu sendiri, proses penanaman komponen-komponen pada PCB. Alurnya mulai dari awal proses SMT dilakukan di PT. SMT Solution Indonesia setelah selesai kemudian barang dikirim ke Value Plus yaitu perusahaan yang bekerja sama dengan PT. SAMSUNG, kemudian barang yang telah dikirim, dikirim kembali ke PT SAMSUNG pusat yang bertempat di Korea. Di PT. SAMSUNG pusat barang di cap kemudian dikirim kembali ke PT. SEIN (SAMSUNG Elektronik Indonesia) yang betempat di

Cikarang, Indonesia yang kemudian akan dirakit menjadi sebuah produk yang siap jual di pasaran.

Sampai akhirnya pada tahun 2007 PT. SMT Solution Indonesia menjalin kerja sama langsung dengan PT. SAMSUNG pusat dalam menangani proses SMT itu sendiri dan kerja sama tersebut sudah terjalin sampai pada saat ini.

2.1.2 Logo Instansi



Gambar 2.1 Logo SSI

2.1.3 Badan Hukum Instansi

Badan hukum PT SMT Solution Indonesia adalah Perseroan Terbatas. Perseroan terbatas merupakan badan usaha dan besarnya modal perseroan tercantum dalam anggaran dasar. Kekayaan perusahaan terpisah dari kekayaan pribadi pemilik perusahaan sehingga memiliki harta kekayaan sendiri. Setiap orang dapat memiliki lebih dari satu saham yang menjadi bukti pemilikan perusahaan. Pemilik saham mempunyai tanggung jawab yang terbatas, yaitu sebanyak saham yang dimiliki. Apabila utang perusahaan melebihi kekayaan perusahaan, maka kelebihan utang tersebut tidak menjadi tanggung jawab para pemegang saham. Apabila perusahaan mendapat keuntungan maka keuntungan tersebut dibagikan sesuai dengan ketentuan yang ditetapkan.

2.1.4 Struktur Organisasi dan Job Description

Organisasi adalah suatu kesatuan yang teratur beberapa kelompok orang yang saling bekerjasama untuk mencapai tujuan bersama. Dalam organisasi diperlukan adanya bentuk dan susunan struktur organisasi untuk menunjukkan kerangka dan susunan perwujudan pola tetap hubungan–hubungan diantara fungsi–fungsi, bagian–bagian atau posisi–posisi, maupun orang–orang yang menunjukkan kedudukan, tugas dan wewenang serta tanggung jawab yang berbeda dalam organisasi. Adapun unsur–unsur organisasi diantaranya kelompok orang dan tujuan bersama.

Tabel 2.1 Struktur Organisasi

President Direktur	**Sr.Han**
Direktur	**Ys.Na**
General Manager	**Iskandar Isa**

Tabel 2.2 Administrator

Accounting	Eka Diana
Hr/Ga	Risa Juniarti
Service Engineer	Sigit
IT	Sunaryo Wijayanto
Security	M.Sohar
O/S	Herwadi
	Riki Faisal
Utility	Suherman
	Syahroni
	Endang Herman
	Eka Januar surya priadi
Driver	Aang Juhana
	Maman Suherman
	Akam Sukiman
	Apandi
C/S Outsorching	Agus
	Asep
	Daim

2.2 Landasan Teori

2.2.1 **Karakteristik Sistem**

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolahan (*process*) dan sasaran (*objectives*) atau tujuan (*goal*).

1. Komponen Sistem (*components*)

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian-bagian dari sistem. Setiap sistem tidak perduli betapapun kecilnya, selalu mengandung komponen-komponen atau sub sistem-subsistem.

2. Batasan Sistem (*boundary*)

   Batasan sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Batas sustu sistem menunjukan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*environments*)

   Lingkungan Luar Sistem (*environments*) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energy dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*interface*)

   Penghubung Sistem (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran (*output*) dari subsistem menjadi masukan(*input*) untuk subsistem yang lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

   Masukan adalah energi yang dimasukan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenanace input*) dan masukan sinyal (*signal input*). *maintenanace input* adalah energi yang dimasukan supaya sistem dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran.

6. Pengolahan Sistem (*Process*)

   Suatu sistem dapat mempunyai bagian pengolah yang akan mengubah masukan menjadi keluaran.

7. Keluaran Sistem (*Output*)

   Keluaran (*Output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dari sisa pembuangan.

8. Sasaran Sistem

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau sustu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentuka sekali masukan yang dibutuhkan sistem dan keluaran

yang akan dihasilakan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.2.2 Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk dihasilkan informasi. Data yang diolah untuk menghasilkan informasi menggunakan suatu model proses tertentu.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan yang berarti menghasilakan suatu tindakan yang lainnya yang akan membuat sejumlah data kembali. Data tersebut akan ditanggap sebagai input, diproses kembali lewat suatu model dan membentuk suatu siklus. Menurut John Burch disebut dengan siklus informasi (*information cycle*) atau siklus ini disebut juga dengan siklus pengolahan data (*data processing cycles*).

2.2.3 Kualitas Informasi

Kualitas dari suatu informasi tergantung dari tiga hal, yaitu informasi harus akurat, tepat pada waktunya dan relevan, diantaranya:

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan yang dapat merubah atau merusak informasi tersebut.
2. Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat.
3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya.

2.2.4 Analis Sistem

Definisi analisis sistem menurut Jogiyanto (1999) adalah penguraian suatu sistem informasi yang utuh ke dalam bagian-bagian komponen dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya. Adapun langkah-langkah dasar yang harus dilakukan oleh analisis sistem sebagai berikut:

1. *Identify*, yaitu mengidentifikasi masalah.
2. *Understand*, yaitu memahami kerja dari sistem yang ada.
3. *Analyze*, yaitu menganalisis sistem.
4. *Report*, yaitu membuat laporan hasil analisis.

2.2.5 Perangkat Dalam Pengembangan Sistem

Perangkat yang digunakan dalam suatu metodologi umumnya berupa suatu gambar atau diagram atau grafik, misalnya *Data Flow Diagram* (DFD). Penggunaan diagram atau gambar ini dipandang lebih mengenal dan lebih dimengerti. *Tools* yang digunakan juga ada yang tidak berupa gambar atau grafik seperti misalnya *data dictionary*.

2.2.5.1 *Data Flow Diagram* (DFD)

Menurut Andri Kristanto 2003, *Data Flow Diagram* (DFD) dapat didefinisikan sebagai suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Melalui suatu teknik analisa data terstruktur yang disebut *Data Flow Diagram* (DFD) analis sistem dapat merepresentasikan proses-proses data di dalam organisasi. *Data Flow*

Diagram memiliki empat kelebihan utama melalui penjelasan naratif mengenai cara data-data berpindah disepanjang sistem yaitu :

1. Kebebasan dari menjalankan implementasi teknis sistem yang terlalu dini.
2. Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
3. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui *Data Flow Diagram* (DFD).

   Menganalisis sistem yang diajukan untuk menentukan apakah data-data dan proses yang diperlukan sudah ditetapkan.

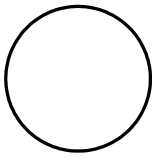
2.2.5.1.1 Definisi *Data Flow Diagram* (DFD)

DFD atau diagram alir data adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan i nteraksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Dalam menggambarkan sistem perlu dilakukan pembentukan simbol, berikut ini simbol-simbol yang digunakan dalam DFD antara lain :

a. Simbol Proses

   Menggambarkan bagaimana suatu input ditransformasikan menjadi output. Proses menggambarkan apa yang dilakukan sistem.

   Simbol :



b. imbol *External Entity*

Sering dikenal dengan sumber, tujuan, terminator. *External entity* adalah bagian luar sistem yang dapat men-*supply* input kedalam sistem dan menggunakan output dari sistem. *External entity* dapat berupa orang/sekelompok orang atau bagian/departemen/pemakai sistem.

Simbol :



c. *Data Store* atau Penyimpanan data

Menggambarkan media penyimpanan data dalam sistem yang berisi data yang digunakan sistem, digunakan sebagai sarana mengumpulkan dan membaca data. *Data Store* dapat merupakan file/database, arsip/catatan manual, lemari/kotak, agenda/buku.

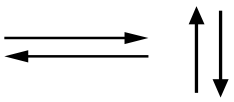
Simbol :



d. Simbol *Data Flow* atau Aliran data

Menggambarkan gerakan paket data/informasi.

Simbol :



2.2.5.1.2 Levelisasi DFD

Didalam DFD terdapat proses pengubahan input menjadi output. Untuk mempermudah pembacaan DFD dibuat bertingkat / *level* dari mulai level 0 sampai dengan *level* yang diperlukan diantaranya yaitu:

1. Diagram Konteks

   Diagram Konteks merupakan *level* tertinggi dari DFD, yang memperlihatkan sistem sebagai sebuah proses. Tujuannya adalah memberikan pandangan umum sistem. Diagram Konteks memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya. Ada *External Entity* yang memberikan masukan (*input*) dan ada pihak yang menerima keluaran (*output*) dari sistem.

2. Diagram Zero

   Diagram yang berada satu *level* dibawah Diagram Konteks yang menggambarkan proses-proses utama dari sistem.hal yang digambarkan dalam diagram *Zero* adalah proses utama dari sistem serta hubungan *terminator* atau entitas proses, *data flow* dan *data store*.

3. Level Diagram (*Diagram Level* n)

   Diagram *level* n merupakan hasil dekomposisi dari diagram *zero*, yang menjelaskan proses secara lebih terperinci. Turunan langsung dari diagram *zero* dinamakan Diagram Level 1. Dan apabila Diagram level 1 dapat diuraikan lagi maka akan terbentuk diagram level 2, dan seterusnya.

2.2.5.2 *Entity Relationship Diagram* (ERD)

Entity Relationship Diagram adalah salah satu model atau teknik pendekatan yang dapat menyatakan atau mengambarkan hubungan antara *entity* di dalam sebuah sistem di mana sebuah hubungan tersebut dinyatakan sebagai *one to one*, *one to many*, *many to one* dan *many to many.*

Entity Relationship Diagram memiliki komponen-komponen utama, yaitu sebagai berikut:

A. Entitas (*Entity*) Entitas adalah sebuah barang atau objek yang dapat dibedakan dari objek lain. Entitas direpresentasikan dengan empat persegi panjang.

B. Atribut (*Attribute*) Atribut adalah properti deskriptif yang dimiliki oleh setiap anggota dari himpunan entitas. Atribut direpresentasikan dengan bentuk *elips*.

C. Relasi (*Relationship*) Relasi adalah hubungan antara suatu himpunan entitas dengan himpunan entitas yang lainnya. Relasi direpresentasi-kan dengan jajaran genjang.

D. Kardinalitas (*Cardinality*) Kardinalitas adalah angka yang menunjukkan banyaknya kemunculan suatu objek terkait dengan kemunculan objek lainnya pada suatu relasi. Jenis-jenis kardinalitas yaitu sebagai berikut :

    a. *One to One.* Hubungan *one to one* adalah antara *file* pertama dengan *file* kedua adalah satu banding satu.

    b. One to Many. Hubungan *one to many* adalah sebuah relasi merupakan tahap dimana hubungan antara *file* pertama dengan

file kedua adalah satu banding banyak.

a. *Many to One.* Hubungan *many to one* adalah sebuah relasi merupakan tahap dimana hubungan antara *file* pertama dengan *file* kedua adalah banyak banding satu.
b. *Many to Many.* Hubungan *many to many* adalah sebuah relasi antara suatu *file* satu dengan *file* yang kedua memepunyai relasi banyak berbanding banyak.

2.2.5.3 Kamus Data

Menurut Kendal dan Kendal, 2003, Kamus data atau data dictionary adalah suatu aplikasi khusus dari jenis kamus – kamus yang digunakan sebagai referensi kehidupan setiap hari. Kamus data merupakan hasil referensi data mengenai data (maksudnya, metadata), suatu data yang disusun oleh penganalisis sistem untuk membimbing mereka selama melakukan analisis dan disain. Sebagai suatu dokumen, kamus data mengumpulkan dan mengkoordinasi istilash – istilah data tertentu, dan menjelaskan apa arti setiap istilah yang ada.

Kamus data berisi:

1. Nama alir data
2. Deskripsi atau penjelasan
3. Stuktur data

2.2.5.4 Basis Data

2.2.5.4.1 Definisi Basis Data

Menurut Fatansyah, (1999) Basis data dapat didefinisikan sebagai berikut:

a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
b. Kumpulan data yang saling berhubungan yang disimpan secara bersama-sama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu untuk memenuhi berbagai kebutuhan.

2.2.5.4.2 Kegunaan Basis Data

Penyusunan basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data yaitu:

A. Redudansi dan inkonsistensi data
   1. Redudansi adalah penyimpanan dibeberapa tempat untuk data yang sama, dan ini akan mengakibatkan pemborosan ruang penyimpanan.
   2. Inkosistensi (tidak konsisten) adalah penyimpanan data yang sama berulang-ulang dibeberapa file

B. Kesulitan pengaksesan data

C. Isolasi data untuk standarisasi.

D. *Multi User* (banyak pemakai)

E. Masalah Keamanan (*security*)

F. Masalah Integritas dan masalah data independence (kebebasan data)

2.2.5.5 Pengelolaan Sistem Basis Data

Pengelolaan basis data atau *Database Management System* (DBMS) secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak atau sistem yang khusus atau spesifik. Sistem ini yang akan menentukan bagaimana data diorganisasikan, disimpan, diubah dan diambil kembali. Disamping itu

sistem ini juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama, pemaksaan keakuratan atau konsistensi data, dan sebagai prangkat lunak yang menangani semua pengaksesan database. Mempunyai fasilitas membuat, mengakses, memanipulasi, dan memelihara basis data.

Perangkat lunak yang termasuk DBMS misalnya *dBase II+, dBase IV, FoxBase, Rbase, MS-Access, dan Borland-Paradox atau Borland-Interbse, MSSQL, SQL Server, CA-Open Ingres, Oracle, Informix* dan *Sybase.*

Fungsi DBMS adalah sebagai berikut :

1. *Data definition*, DBMS dapat mengolah pendefinisian data.
2. *Data manipulation*, DBMS dapat menangani permintaan dari pemakai untuk mengakses data.
3. *Data security and integrity*, DBMS dapat memeriksa keamanan dan integritas data yang didefinisikan oleh DBA.
4. *Data recovery and concurency*, DBMS dapat menangani kegagalan-kegagalan pengaksesan database yang dapat disebabkan oleh kesalahan sistem, kerusakan disk, dan sebagainya.
5. *Data dictionary*, DBMS dapat menyediakan *data dictionary*.
6. *Performance*, DBMS dapat menangani unjuk kerja dari semua fungsi seefisien mungkin.

Sebuah DBMS umumnya memiliki sejumlah komponen fungsional (modul) seperti :

1. *File manager*, yang mengelola ruang dalam disk dan struktur data yang dipakai untuk merepresentasikan informasi yang tersimpan dalam disk.

2. *Database manager*, yang menyediakan interface antara data low-level yang ada di basis data dengan program aplikasi dan query yang diberikan ke sistem.
3. *Query processor*, yang menterjemahkan perintah-perintah dalam query language ke perintah low-level yang dapat dimengerti oleh database manager.
4. *DML precompiler*, yang mengkonversi perintah DML dalam sebuah program aplikasi prosedur normal dalam bahasa induk.
5. *DDL compiler*, yang mengkonversi perintah-perintah DDL ke dalam sekumpulan tabel yang mengandung metadata. Tabel-tabel ini kemudian disimpan dalam kamus data.