BAB II
TINJAUAN PUSTAKA

II.1 Profil Perusahaan

Profil perusahaan adalah untuk mengetahui keadaan di perusahaan diantaranya adalah sejarah perusahaan, visi, misi dan fokus perusahaan, struktur organisasi perusahaan dan deskripsi tugas.

II.1.1 Sejarah Perusahaan

PT Abdi Daya Mitra Abadi (ADMA) adalah perusahaan pelayanan yang mengkhususkan diri pada Penyediaan dan Pengelolaan Tenaga Kerja, Pusat Pendidikan dan Pelatihan Tenaga Kerja, *Out Sourching Management, Man Power Supply* yang secara profesional berdiri sejak 17 September 2012 melalui Akta Notaris Arie Prio Buntoro, SH No.16 Tanggal 17 September 2012 yang mana berdirinya perseroan tersebut adalah untuk menyesuaikan diri dengan undang-undang ketenagakerjaan No.13 Tahun 2003 yang memposisikan kedudukan sebagai perusahaan *Out Sourching* dengan jelas yang mana dalam salah satu pasal mensyaratkan perusahaan *Out Sourching* harus berbadan hukum Perseroan Terbatas (PT). PT Abdi Daya Mitra Abadi (ADMA) bertempat di Jalan KH Ahmad Dahlan (Jl. Banteng) No. 73C Bandung [1].

PT Abdi Daya Mitra Abadi (ADMA) sangat peduli untuk memberikan dan membuka lapangan pekerjaan kepada siapa saja yang membutuhkan pekerjaan sesuai dengan keahlian dan tingkat pendidikannya. Dalam rangka pengembangan usaha PT Abdi Daya Mitra Abadi (ADMA) telah mendirikan beberapa unit usaha yang hingga kini telah tumbuh berkembang seiring waktu dan telah pula mengadakan kontrak kerjasama dengan berbagai instansi dan perusahaan yang membutuhkan jasa pengadaan dan pengelolaan tenaga kerja dan pengamanan sekaligus [1].

Sumber daya manusia, baik sektor industri, properti dan lain sebagainya pada saat ini merupakan suatu kebutuhan yang sangat mutlak dan untuk menjawab tantangan tersebut, maka PT Abdi Daya Mitra Abadi (ADMA) selalu

mengedepankan pelayanan yang terbaik agar para pengguna jasa dapat merasakan manfaat dan keuntungan yang maksimal. PT Abdi Daya Mitra Abadi (ADMA) didukung dengan tenaga kerja yang terlatih, berdedikasi, penuh rasa tanggung jawab dan profesional, ditambah pula instruktur serta sistem kerja yang telah disusun berdasarkan kondisi yang ada di lingkungan pengguna jasa. PT Abdi Daya Mitra Abadi (ADMA) sangat siap memenuhi kebutuhan akan pelayanan kepada siapapun, baik untuk kebutuhan pribadi, instansi pemerintahan ataupun swasta, serta perumahan. Kapanpun, dimanapun dan dalam kondisi seperti apapun [1].

II.1.2 Visi

Visi PT Abdi Daya Mitra Abadi (ADMA) yaitu menjadi perusahaan penyedia dan pengelola sumber daya manusia yang professional, terpercaya, terpilih dan terpandang [1].

II.1.3 Misi

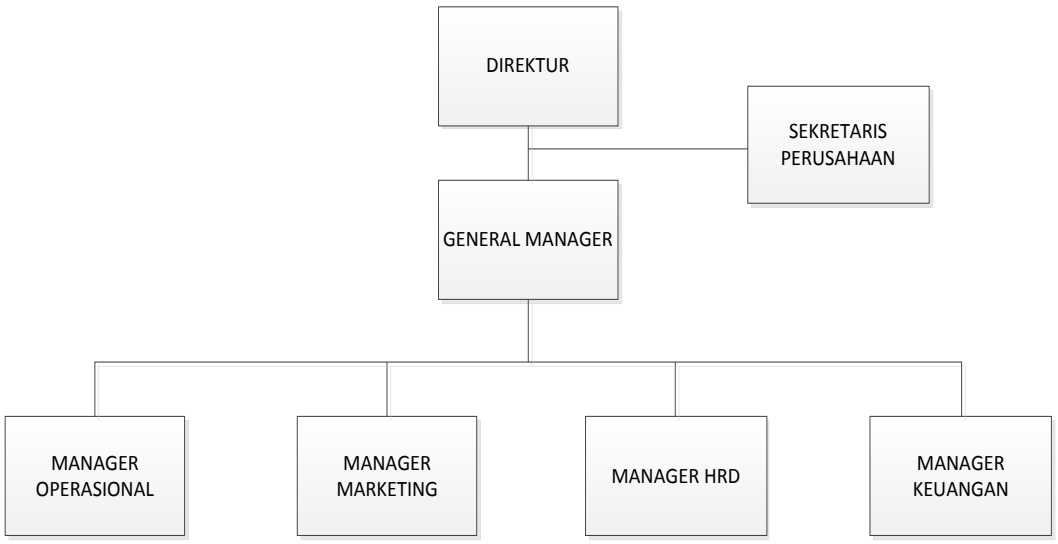
Misi PT Abdi Daya Mitra Abadi (ADMA) yaitu memberikan nilai tambah lebih bagi pemegang saham, pengguna jasa, karyawan dan masyarakat serta mendukung terbukanya lapangan pekerjaan baru. Melalui sumber daya manusia yang professional, kami senantiasa membangun kemitraan, semua untuk yang terbaik dan yang terbaik untuk semua [1].

II.1.4 Fokus

Fokus PT Abdi Daya Mitra Abadi (ADMA) yaitu menggunakan seluruh kompetensi perusahaan secara optimal untuk meningkatkan nilai tambah perusahaan [1].

II.1.5 Struktur Organisasi

Struktur organisasi merupakan hal yang penting agar setiap pegawainya dapat bekerja sesuai dengan *job* d*escription*nya masing-masing. Struktur organisasi PT ADMA dapat dilihat pada gambar II.1.



Gambar II.1. Struktur Organisasi PT ADMA

II.1.6 Deskripsi Tugas

Deskripsi tugas digunakan untuk mengetahui tugas, wewenang dan tanggung jawab dari masing-masing bagian. PT ADMA memiliki satu sekretaris perusahaan, satu manager umum yang bertanggung jawab atas kinerja dari manager operasional, manager marketing, manager HRD, dan manager keuangan. Bagian operasional memiliki dua staf, bagian marketing memiliki dua staf, bagian HRD memiliki satu staf, dan bagian keuangan memiliki satu staf. Deskripsi tugas yang ada di PT Abdi Daya Mitra Abadi (ADMA) adalah sebagai berikut.

1. Direktur

   Memimpin perusahaan dengan menerbitkan kebijakan-kebijakan perusahaan. Memilih, menetapkan, mengawasi tugas karyawan serta menyetujui anggaran tahunan perusahaan.
2. Sekretaris Perusahaan

   Bertanggung jawab atas administrasi efisien perusahaan, khususnya berkaitan dengan memastikan kepatuhan terhadap persyaratan hukum dan peraturan, serta memastikan bahwa keputusan Direktur dilaksanakan.

3. General Manager

   Bertanggung jawab atas seluruh bagian atau fungsional perusahaan.

4. Manager Operasional

   Bertanggung jawab dan membuat perencanaan kegiatan penyegaran, pembinaan, jadwal sidak maupun jadwal pembagian wilayah patroli untuk staff operasional. Melakukan koordinasi yang berhubungan dengan kegiatan satpam dilapangan dengan klien.

5. Manager Marketing

   Bertanggung jawab dan membuat perpanjangan kontrak kerja sama, merancang strategi pemasaran, melakukan promosi, pengawasan dan evaluasi kinerja bagian marketing.

6. Manager HRD

   Bertanggung jawab atas penerimaan anggota baru, dokumen-dokumen persyaratan, ijin perusahaan, filling/pengarsipan administrasi karyawan maupun anggota, melakukan pengawasan dan evaluasi di bangian HRD.

7. Manager Keuangan

   Bertanggung jawab dalam melakukan perencanaan, pengangaran, pemeriksaaan, pengelolaan, pengendalian, pencairan dana, penyimpanan dana yang dimiliki perusahaan dan pertanggungjawaban keuangan.

II.2 Landasan Teori

Teori-teori yang akan diuraikan adalah teori mengenai konsep dasar sistem informasi, *customer relationship management* (CRM), basis data, perangkat lunak pendukung, dan pemodelan data.

II.2.1 Konsep Dasar Sistem Informasi

Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas sebuah komponen fungsional (dengan satu fungsi/tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses/pekerjaan tertentu. Informasi adalah data yang telah diolah menjadi sebuah bentuk yang

berarti bagi penerimanya dan bermanfaat dalam pengambian keputusan saat ini atau saat mendatang [2].

Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada para pemakai [2].

II.2.2 Sistem Informasi

Sistem informasi dibentuk dari dua kata yaitu sistem dan informasi. Sistem adalah hubungan satu unit dengan unit-unit lainnya yang saling berhubungan satu sama lain dan tidak dapat dipisahkan serta menuju suatu kesatuan untuk mencapai tujuan tertentu, apabila terdapat gangguan atau kendala maka tujuan yang tercapai pun akan terganggu [3].

Kualitas informasi dapat dipengaruhi oleh enam hal, yaitu :

a. Relevan

b. Akurat

c. Tepat waktu

d. Ekonomis

e. Efisien

f. Dapat dipercaya

Setelah mendapatkan pengertian sistem dan informasi maka sistem informasi merupakan sekumpulan subsistem yang saling berhubungan, berkumpul bersama–sama dan membentuk satu kesatuan, saling berinteraksi dan bekerja sama antara bagian satu dengan yang lainnya dengan cara-cara tertentu untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi sebagai dasar bagi pengambilan keputusan yang berguna dan mempunyai nilai yang dapat dirasakan akibatnya baik pada saat itu juga maupun di masa mendatang, mendukung kegiatan operasional, manajerial, dan strategis organisasi, dengan memanfaatkan beberapa sumber daya yang tersedia bagi fungsi tersebut guna mencapai tujuan [4].

II.2.2.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah, dan sasaran atau tujuan [2].

1. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya saling bekerjasama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai karakteristik dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem (*Boundary*)

   Batas Sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini dapat dipandang sebagai satu kesatuan dan menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

   Lingkungan luar dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*Interface*)

   Merupakan suatu media yang menghubungkan antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. Keluaran dari satu subsistem akan menjadi masukan untuk subsistem lainnya melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

   Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem yang berupa masukan perawatan (*maintenance input*) agar sistem tersebut dapat beroperasi dan masukan sinyal yang diproses untuk menghasilkan keluaran. Sebagai contoh di dalan sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputer dan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

   Merupakan hasil dari energi yang telah diproses dan dapat diklasifikasikan antara lain menjadi *output* yang langsung digunakan, *output* yang bisa dijadikan *input* untuk subsistem yang lain atau kepada supra sistem serta *output* yang bagi sistem tidak digunakan lagi (sisa pembuangan) tapi bagian sistem lain sangat berguna.

7. Pengolah Sistem (*Process*)

   Suatu sistem dapat mempunyai bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merupakan masukan menjadi keluaran. Sistem akuntansi akan megolah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan yang dibutuhkan oleh manajemen.

8. Sasaran dan Tujuan

   Suatu sistem dikatakan berhasil jika mengenai sasaran atau tujuannya, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

II.2.2.2 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lainnya. Karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan kedalam beberapa sudut pandang [2].

a. Sistem diklasifikasikan sebagai sistem alamiah dan sistem buatan manusia.

   Sistem alamiah adalah sistem yang terjadi karena proses alam tidak dibuat oleh manusia (ditentukan dan tunduk kepada kehendak sang pencipta alam). Sedangkan sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin disebut dengan *human-machine system* atau ada yang menyebut dengan *machine system*.

b. Sistem diklasifkasikan sebagai sistem tertutup dan sistem terbuka.

   Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak berpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan pihak luarnya. Secara teoritis sistem tertutup ini ada, tetapi kenyatannya tidak ada sistem yang benar-benar tertutup, yang ada hanya *relatively closed system* (secara relatif tertutup, tidak benar-benar tertutup). Sistem terbuka adalah sistem yang berhubungan dan tepengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lain. Karena sistem sifatnya terbuka dan terpengaruh lingkungan luarnya, maka suatu sistem harus mempunyai suatu pengendalian yang baik. Sistem yang baik harus dirancang sedemikian rupa, sehingga secara relatif tertutup karena sistem tertutup akan bekerja secara otomatis dan terbuka hanya untuk pengaruh yang baik saja.

II.2.2.3 Customer Relationship Management (CRM)

CRM adalah singkatan dari *customer relationship management*. CRM memiliki arti sebagai suatu konsep manajemen secara menyeluruh yang berpusat pada pelanggan sebagai faktor utama untuk meningkatkan keunggulan bersaing dan bertujuan memperoleh tingkat kepuasan pelanggan secara maksimum yang akan membawa keuntungan bagi perusahaan dengan biaya yang relatif lebih rendah. CRM adalah strategi bisnis dengan menerapkan teknologi informasi pada perusahaan secara komprehensif, handal, dan terintegrasi berdasarkan sudut

pandang pelanggan, dengan begitu semua proses dan interaksi saling membantu mempertahankan dan mengembangkan hubungan yang bermanfaat [6].

Customer Relationship Management (CRM) merupakan strategi yang diimplementasikan secara luas untuk mengatur hubungan antara perusahaan dengan pelanggan, *client*, dan calon pelanggan. CRM menggunakan suatu teknologi untuk mengorganisasi, otomasi, dan sinkronisasi semua proses bisnis, terutama kegiatan–kegiatan penjualan; tetapi juga kegiatan–kegiatan pemasaran, pelayanan pelanggan dan dukungan teknis. Prinsip dalam CRM ialah memperoleh pelanggan, mempertahankan pelanggan, dan memberikan pelayanan yang berkualitas sehingga pelanggan merasa puas dan menjadi loyal terhadap perusahaan. Biasanya CRM mengacu pada tiga hal, pertama, CRM merupakan bidang yang berfokus pada pelanggan seperti pemasaran penjualan, dan dukungan pelanggan. Kedua, CRM menggunakan alat yang memiliki fungsi sebagai *sales force automation* (SFA). Ketiga, CRM merupakan proses yang mengelola hubungan dengan pelanggan.

II.2.2.4 Tujuan CRM

Tujuan dari CRM adalah [11]:

1. Menggunakan hubungan dengan pelanggan untuk meningkatkan keuntungan perusahaan.
2. Menggunakan informasi untuk memberikan pelayanan yang memuaskan.
3. Mendukung proses penjualan berulang kepada pelanggan.

Keuntungan menggunakan CRM adalah servis yang lebih cepat, mengurangi harga, memperbesar keuntungan, mempunyai rasa memiliki, meningkatkan koordinasi team, tingkat kepuasan pelanggan lebih tinggi, meningkatkan loyalitas pelanggan.

II.2.2.5 Manfaat CRM

CRM sebagai sebuah sistem dalam perusahaan diharapkan dapat memberikan keuntungan yang lebih besar terhadap perusahaan dalam mencapai tujuan perusahaan. Tujuh keuntungan utama CRM adalah :

a. Meningkatkan kemampuan untuk mentargetkan keuntungan yang dapat diraih dari pelanggan.
b. Mengintegrasikan penawaran lintas *channel*.
c. Meningkatkan efisiensi dan efektifitas usaha penjualan.
d. Menyajikan pesan pemasaran secara individu.
e. Meningkatkan pelayanan dan produk.
f. Meningkatkan efisiensi dan efektifitas pelayanan kepada pelanggan.
g. Meningkatkan harga.

Adapun keuntungan dari penerapan CRM adalah sebagai berikut :

a. Membantu perusahaan dalam memberikan akses terhadap departemen pemasaran untuk mengidentifikasi dan menargetkan pelanggan potensial yang dimilikinya, mengelola kampanye pemasaran, dan menempatkan pemimpin dalam kelompok penjual.
b. Membantu organisasi untuk meningkatkan penjualan melalui telepon, akun pelanggan, dan pengelolaan penjualan melalui pemaksimalan proses berbagi informasi melalui lintas fungsional perusahaan.
c. Mempersilahkan formasi hubungan individual dengan pelanggan, dengan tujuan untuk meningkatkan kepuasan pelanggan, identifikasi terhadap pelanggan yang paling menguntungkan, dan memberikan mereka level pelayanan yang terbaik.
d. Menyajikan kepada karyawan, informasi dan proses mudah untuk mengenal pelanggannya dan mengidentifikasi kebutuhan pelanggan dan membangun hubungan yang efektif antara perusahaan, pelanggan, dan rekanan distributor.

II.2.2.6 Jenis-jenis CRM

Jenis-jenis CRM adalah [11]:

a. Strategis (*Strategic*)

Perspektif strategis berfokus pada pengembangan budaya bisnis berorientasi pelanggan. Budaya ini diterapkan untuk dapat memenangkan dan menjaga pelanggan, dengan memberikan pelayanan yang lebih baik dibandingkan pesaing. Budaya ini juga mengharuskan perusahaan untuk

mengetahui lebih banyak mengenai pelanggan supaya dapat melayani pelanggan secara lebih intim untuk menghasilkan keuntungan lebih bagi perusahaan.

b. Operasional (*Operational*)

Perspektif operasional berfokus pada otomatisasi dan meningkatkan tatap muka dengan pelanggan dan proses bisnis penunjang bagi pelanggan. Fokus otomatisasi ini memungkinkan adanya integrasi antara fungsional perusahaan seperti fungsi pemasaran, penjualan, dan pelayanan.

c. Analitis (*Analytical*)

Pada perspektif ini berfokus pada proses menangkap, menyimpan, mengekstrak, mengintegrasi, memproses, menerjemahkan, mendistribusikan, menggunakan dan melaporkan data yang terkait dengan pelanggan untuk meningkatkan nilai pelanggan dan perusahaan. Data yang digunakan untuk menganalisis informasi mengenai pelanggan seperti data pemasaran, data pembayaran, data pengiriman, data pribadi pelanggan, data interaksi pelanggan dalam sistem, dan data transaksi pelanggan.

d. Kolaborasi (*Collaborative*)

Fokus kolaborasi pada CRM memungkinkan perusahaan yang bekerja pada sebuah *cluster* bisnis atau dalam proses kerja sama dalam rantai pasok untuk dapat berbagi data lintas fungsional dan lintas perusahaan. Persebaran informasi yang terjadi memungkinkan perusahaan untuk dapat mengidentifikasi permasalahan yang terjadi dan mengembangkan pelanggan dengan melayani mereka lebih baik.

II.2.2.7 Tahapan Dalam CRM

Pengimplentasian konsep CRM yang baik adalah mengorganisasikan proses CRM di sekitar pelanggan dan tidak hanya pada fungsi internal perusahaan. Terdapat tiga tahapan dalam CRM, yaitu [5]:

1. *Acquire*

Acquire merupakan suatu tahap mengakuisisi pelanggan. Melalui tahap ini dapat memberikan sebuah penilaian kepada perusahaan dengan menilai

kesan pertama terhadap perusahaan. Apabila pelanggan memiliki penilaian yang baik terhadap perusahaan, maka transaksi akan dapat berjalan dengan baik. Mengakuisisi pelanggan baru dengan melakukan promosi atas paket dan jasa yang ditawarkan.

2. *Enhance*

   Enhance merupakan suatu proses dimana perusahaan membangun hubungan yang baik dengan pelanggan. Dengan mendengarkan keluhan dari pelanggan dan menerima saran dari pelanggan merupakan salah satu cara yang efektif dalam membangun hubungan dengan pelanggan.

   Dalam tahap ini ada dua hal yang harus diperhatikan:

   a. *Cross selling*, yaitu merupakan salah satu strategi dalam mengidentifikasi pelanggan guna menambah penjualan. Misalnya, menawarkan paket pelengkap kepada pelanggan.
   b. *Up Selling*, yaitu merupakan strategi meningkatkan penjualan paket melalui penambahan kuantitas paket. Misalnya, menawarkan paket yang sama dengan kuantitas yang lebih besar.

3. *Retain*

   Retain merupakan tahap dimana perusahaan berusaha mampertahankan hubungan dengan pelanggan, dengan kata lain mendapatkan pelanggan dan kemudian memelihara hubungan dengan mereka. Agar dapat menjalin hubungan dengan terus-menerus dan mempertahankan hubungan dengan pelanggan, maka perusahaan harus menggunakan pandangan yang menggunakan pelanggan dengan memberikan pelayanan dan aplikasi pendukung.

II.2.2.8 Framework CRM

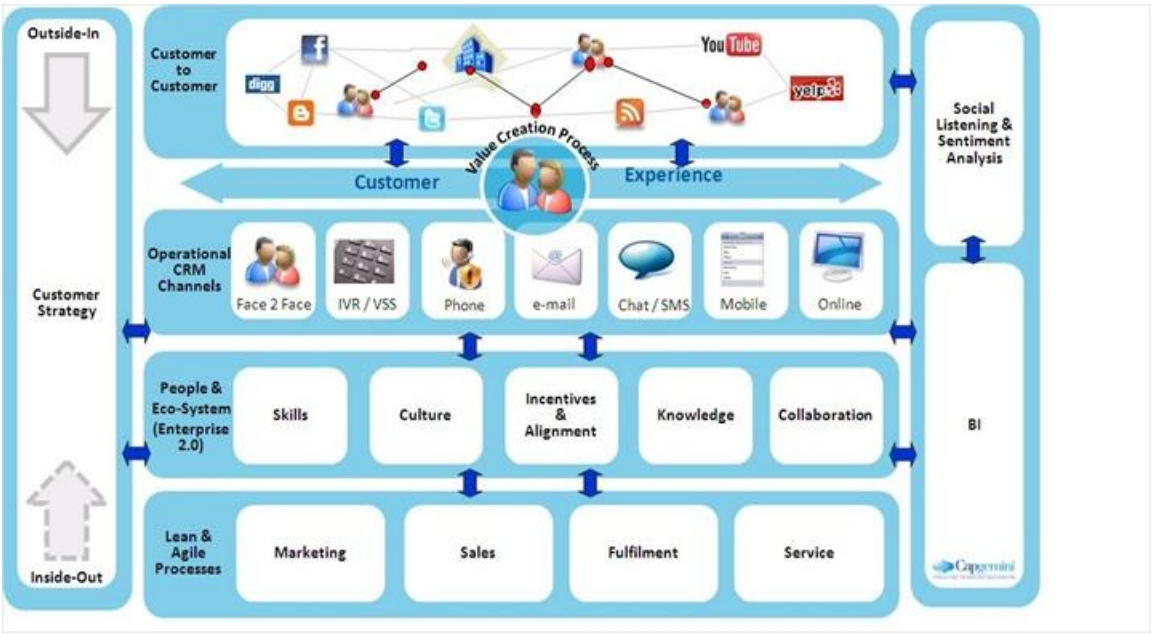
Framework adalah kerangka kerja yang dapat digunakan untuk mengembangkan sistem CRM. Inti dari revolusi teknologi informasi CRM adalah teknologi berbasis web dimana perusahaan diberikan kesempatan untuk memilih bagaimana perusahaan berinteraksi dengan pelanggan. Teknologi berbasis web ini memungkinkan perusahaan membangun hubungan yang baik dengan pelanggan

daripada sebelumnya dengan cara manual. Penggunaan web memiliki kemampuan untuk merespon langsung permintaan pelanggan dan menyediakan kebutuhan pelanggan dengan interaktif selain itu perusahaan dapat memanfaatkan biaya yang rendah dengan layanan yang berkualitas [6].

Penelitian ini mengusulkan sebuah kerangka kerja konseptual yang bertujuan untuk menerapkan CRM berhasil berdasarkan, studi sebelumnya, dan pengalaman. *Framework* kerangka mengintegrasikan tiga konstruksi utama yaitu fase implementasi CRM, komponen CRM utama, dan CRM sukses faktor. Kerangka kerja ini dimulai dengan tahap pra-implementasi dimana organisasi harus mendapatkan dukungan dari manajemen puncak untuk menerapkan CRM sebelum melibatkan dalam pelaksanaan proses yang sebenarnya [6].

Berdasarkan komitmen yang memadai oleh manajemen puncak proses bisa dilanjutkan untuk menilai kesiapan organisasi dalam hal manusia, teknologi, dan organisasi (proses) kemampuan untuk menerapkan CRM, Oleh karena itu, strategi CRM harus dikembangkan, selaras, dan dikomunikasikan kepada staf. Panah mencerminkan aliran proses implementasi. Strategi CRM harus dimasukan ke dalam tindakan dengan menerapkan proyek CRM berfokus pada faktor-faktor keberhasilan dalam setiap komponen dari CRM. Akibatnya, pelaksanaan setiap faktor keberhasilan harus diukur untuk memastikan keberhasilan pelaksanaan proyek CRM [6].

Langkah terakhir kerangka berfokus pada mengukur dampak implementasi CRM secara finansial dan pada pemasaran kinerja untuk memastikan mewujudkan tujuan dari strategi CRM dan untuk menilai pengaruh lainnya mungkin faktor eksternal. Tanda panah naik mencerminkan proses umpan balik dan pembelajaran dari pelaksanaan yang didasarkan pada pengukuran faktor kesuksesan CRM dan mengukur kinerja proyek CRM setelah sepenuhnya diimplementasikan sebagai proses. Berikut dapat dilihat pada gambar II.2.



Gambar II.2. *Framework* CRM [6]

Penjelasan *layer* pada *framework* CRM :

1. ***Layer Customer Strategy*** : *Layer* awal dimana perusahaan menentukan strategi untuk menjangkau pelanggan. *Layer* ini memiliki atribut *inside-out* dan *outside-in* . *Inside-out* yang berarti strategi yang berasal dari perusahaan itu sendiri dan diterapkan kepada pelanggan, dan *outside-in* yang berarti strategi yang diterapkan oleh perusahaan berdasarkan saran atau kritik yang diberikan oleh pelanggan
2. ***Layer Customer to Customer*** : *Layer* ini menjelaskan bagaimana cara pelanggan berbagi informasi tentang perusahaan kepada orang lain atau perusahaan lain.
3. ***Layer Customer Experience*** : *Layer* yang merupakan akumulasi pengalaman pelanggan terhadap perusahaan. Pada tahap ini dalam sistem akan menampilkan pelanggan mana yang telah menjalin hubungan kerjasama dengan pihak perusahaan dan memiliki loyalitas tinggi dengan ketentuan dengan ketentuan pelanggan sudah memenuhi masa periode tertentu.
4. ***Layer Operational CRM Channels*** : *layer* ini merupakan tahapan berkomunikasi dan berinteraksi antara perusahaan dan pelanggan.

5. ***Layer People and Eco-system*** : menunjukan SDM yang kompeten untuk menjalankan sistem CRM yang akan dibangun.
6. ***Layer Lean & Agile Processes*** *:* Layer ini berhubungan dengan semua proses bisnis yang ada kaitannya dengan pelanggan.

II.2.3 Basis Data

Basis data dapat dipahami sebagai suatu kumpulan data yang terhubung (*interrelated data*) yang disimpan secara bersama–sama pada suatu media, tidak perlu suatu kerangkapan data (walaupun ada harus seminimal mungkin dan terkontrol (*controlled redundancy*)), data disimpan dengan cara–cara tertentu sehingga mudah digunakan atau ditampilkan kembali. Data dapat digunakan oleh satu atau lebih program–program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan sedemikianrupa sehingga proses penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.

Basis data (*database*) sendiri dapat didefiniskan sebagai berikut :"Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan pada perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya."[7].

II.2.4 Perangkat Lunak Pendukung

Perangkat lunak pendukung merupakan sistem yang terkait untuk mendukung agar sistem berjalan sebagaimana mestinya.

II.2.4.1 MySQL

MySQL adalah multi *user database* yang menggunakan bahasa *Strucktured Query Language* (SQL). MySQL dalam operasi *client-server* melibatkan *server daemon* MySQL disisi *server* dan berbagai macam program serta *library* yang berjalan disisi *client*. MySQL mampu menangani data yang cukup besar. Perusahaan yang mengembangkan MySQL yaitu TcX, mengaku menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 *Gigabyte* data.

SQL adalah bahasa standar yang digunakan untuk mengakses *database server*. Bahasa ini pada awalnya dikembangkan oleh IBM, namun telah diadopsi dan digunakan sebagai standar industri. Dengan menggunakan SQL, proses akses *database* menjadi lebih *user-friendly* dibandingkan dengan menggunakan *dBASE* atau *Clipper* yang masih menggunakan perintah-perintah pemrograman.

MySQL merupakan *software database* yang paling populer di lingkungan *Linux*, kepopuleran ini karena ditunjang performa *query* dari *database*nya yang saat ini bisa dikatakan paling cepat dan jarang bermasalah. MySQL ini juga sudah dapat berjalan pada lingkungan *Windows* [8].

II.2.4.2 Wampserver

Wampserver adalah sebuah aplikasi *open source* untuk pengembangan sebuah web di *Windows*. *Wampserver* memungkinkan kita untuk mengembangkan web dengan Apache2, PHP dan *database* MySQL. Dengan kata lain, *wampserver* digunakan untuk membuat *web server* lokal pada komputer guna memudahkan perancangan dan pembuatan web sebelum dipublikasikan ke internet atau jaringan lokal (LAN).

WampServer merupakan sebuah *manager service* yang akan menginstal Apache, PHP5, *database* MySQL, PHP*myadmin* dan *SQLitemanager* di komputer. *WampServer* 2.0e dahulu bernama WAMP5.

Wamp adalah sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah *server*. Kegunaan *wamp server* ini untuk membuat jaringan lokal sendiri dalam artian kita dapat membuat *website* secara *offline* untuk masa coba-coba di komputer sendiri. Jadi fungsi dari wamp server itu sendiri merupakan *server website* kita untuk cara memakainya [8].

II.2.4.3 PHP

PHP adalah bahasa pemrograman *server side* yang sudah banyak digunakan saat ini, terutama untuk pembuatan *website* dinamis. Untuk hal-hal tertentu dalam pembuatan web, bahasa pemrograman PHP memang diperlukan, misalnya saja pemroses data yang dikirimkan oleh pengunjung web. Contoh nyata adalah

penggunaanya buku untuk tamu, halaman *login*, dan masih banyak lagi. Agar bisa berjalan dengan baik maka PHP harus dikolaborasikan dengan dengan *database engine*, misalnya saja MySQL, *PostgresSQL*, dan lain-lain [9].

PHP (*Hypertext Preprocessor*) adalah sebuah bahasa *scripting* yang dibundel dengan HTML, yang berjalan disisi *server*. Sebagian besar perintahnya berasal dari bahasa C, *Java* dan *Perl* dengan beberapa tambahan fungsi khusus PHP. Bahasa ini memungkinkan para pembuat aplikasi web yang menyajikan HTML yang dinamis dan interaktif dengan cepat dan mudah, yang dihasilkan *server*. PHP biasa berinteraksi dengan hampir semua teknologi web yang sudah ada. *Developer* biasa menulis sebuah program PHP yang mengeksekusi suatu program CGI di *server* lain. Fleksibilitas ini amat bermanfaat bagi pemilik situs-situs web yang besar dan sibuk, karena pemilik masih menggunakan aplikasi-aplikasi yang sudah terlanjur dibuat dimasa lalu dengan menggunakan CGI, ISAP atau dengan *script* seperti *Perl, awk* atau *Phyton* selama proses migrasi ke aplikasi baru yang dibuat dengan menggunakan PHP.

II.2.5 Pemodelan Data

Pemodelan data merupakan bentuk grafis yang menggambarkan data yang akan terkait dengan sistem.

II.2.5.1 Unifed Modeling Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *Class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET [10].

Pada pemodelan UML terdapat beberapa diagram yaitu *Use Case Diagram, Sequence Diagram, Class Diagram, Activity Diagram.*

1. *Use Case Diagram*

   Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *Use Case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use Case* merupakan sebuah pekerjaan tertentu, misalnya Login ke sistem, mengbuat sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

   Use Case Diagram dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem. Sebuah *Use Case* dapat meng-*include* fungsionalitas *Use Case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *Use Case* yang di-*include* akan dipanggil setiap kali *Use Case* yang meng-*include* dieksekusi secara normal. Sebuah *Use Case* dapat di-*include* oleh lebih dari satu *Use Case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang sama. Sebuah *Use Case* juga dapat meng-*extend Use Case* lain dengan kebutuhannya sendiri. Sementara hubungan generalisasi antar *Use Case* menunjukkan bahwa *Use Case* yang satu merupakan spesialisasi dari yang lain [10].

2. *Sequence Diagram*

   Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

   Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal

dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal. Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi / metode dari *Class*. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah *message.* Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan icon khusus untuk objek *boundary, controller* dan *persistent entity* [10].

3. *Class Diagram*

   Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

   Class diagram menggambarkan struktur dan deskripsi *Class, package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain [10].

   Hubungan antar *Class diagram* dapat dilihat dibawah ini :

   a. Asosiasi, yaitu hubungan statis antar *Class*. Umumnya menggambarkan *Class* yang memiliki atribut berupa *Class* lain, atau *Class* yang harus mengetahui eksistensi *Class* lain. Panah *navigability* menunjukkan arah *query* antar *Class*.
   b. Agregasi, yaitu hubungan yang menyatakan bagian (“terdiri atas..”).
   c. Pewarisan, yaitu hubungan hirarkis antar *Class*. *Class* dapat diturunkan dari *Class* lain dan mewarisi semua atribut dan metode *Class* asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari *Class* yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
   d. Hubungan dinamis, yaitu rangkaian pesan (*message*) yang di-*passing* dari satu *Class* kepada *Class* lain. Hubungan dinamis dapat digambarkan dengan menggunakan *Sequence diagram*.

4. *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya state sebelumnya (*internal processing*). Oleh karena itu *Activity diagram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *Use Case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *Use Case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti state, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa object *swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu [10].