

BAB 2

LANDASAN TEORI

2.1 Teori – Teori Dasar

Untuk membantu melakukan analisa sistem informasi berjalan, maka diperlukan pemahaman mengenai konsep dasar sistem informasi yang meliputi pemahaman akan definisi dari kata yang membentuknya. Berikut ini dijelaskan konsep dasar dari sistem informasi.

2.1.1 Pengertian Sistem

Menurut Mathiassen (2000, p9) Sistem merupakan kumpulan dari komponen yang mengimplementasikan kebutuhan-kebutuhan perancangan, fungsi-fungsi, dan tampilan-tampilan.

Menurut O'Brien (2005), sistem adalah sekumpulan komponen yang saling berhubungan yang bekerja sama untuk mencapai tujuan dengan menerima *input* dan menghasilkan *output* dalam proses transformasi yang teratur.

Didalam sistem terdapat tiga aspek penting yang harus diperhatikan, yaitu:

1. *Input*, merupakan suatu proses pencatatan dan pengumpulan elemen yang masuk ke dalam sistem untuk diproses. Misalnya data bahan baku, data ketenagakerjaan, dan lain – lain.

2. *Processing*, merupakan proses di dalam sistem yang mengubah *input* menjadi *output*. Seperti proses produksi, proses manusia bernafas, atau perhitungan matematika.
3. *Output*, merupakan pemindahan elemen – elemen yang telah di dihasilkan oleh sebuah proses informasi menjadi tujuan akhir yang diharapkan. Seperti penyelesaian produk dan pelayanan jasa.

2.1.2 Pengertian Informasi

Menurut O'Brien (2005), informasi adalah data yang telah diubah menjadi sesuatu yang memiliki arti dan manfaat bagi pengguna secara khusus.

2.1.3 Pengertian Sistem Informasi

Menurut O'Brien dan Marakas (2006, p6), sebuah sistem informasi bisa merupakan kombinasi dari orang – orang, *hardware*, *software* jaringan komunikasi, dan sumber – sumber data yang disimpan, diperoleh, dirubah dan dihilangkan di dalam suatu organisasi. Orang – orang yang sudah bergantung pada sistem informasi untuk berkomunikasi dengan orang lain, dengan menggunakan berbagai macam *hardware*, *software*, jaringan, dan sumber – sumber data.

Menurut Whitten dan Bentley (2007,p6). Sistem informasi adalah suatu pengaturan dari orang, data, proses dan teknologi informasi yang saling berinteraksi untuk mengumpulkan, memproses, menyimpan dan menghasilkan kegunaan yang berupa informasi yang dibutuhkan untuk mendukung suatu organisasi.

2.1.4 Enterprise Resource Planning (ERP)

2.1.4.1 Pengertian Enterprise Resource Planning (ERP)

Menurut Brandy (2006,p2), *Enterprise Resource Planning* (ERP) adalah sebuah sistem yang membantu untuk mengatur proses bisnis seperti pemasaran, produksi, pembelan dan akuntansi dalam sebuah cara yang terintegrasi.

Menurut O'Brien (2004, p194), pengertian *Enterprise Resource Planning* (ERP) adalah tulang punggung perusahaan lintas fungsi yang mengintegrasikan dan mengotomatisasikan banyak proses bisnis internal dan sistem informasi dalam fungsi penjualan dan distribusi, produksi, logistik, akuntansi, dan sumber daya manusia sebuah perusahaan.

Menurut Dhewanto dan Falahah (2007, p2). ERP singkatan dari 3 element kata, yaitu *enterprise* (perusahaan/organisasi), *resource* (sumber daya), dan *planning* (perencanaan). Tiga kata ini mencerminkan sebuah konsep yang berujung kepada kata kerja, yaitu "*planning*" yang berarti bahwa ERP menekankan kepada aspek perencanaan.

Menurut Wijaya dan Darudianto (2009, p27). *Enterprise Resource Planning* (ERP) merupakan konsep untuk merencanakan dan mengelola sumber daya perusahaan, yaitu berupa paket aplikasi program terintegrasi dan multi modul yang dirancang untuk melayani dan mendukung berbagai fungsi dalam perusahaan (*to serve and support multiple business functions*), sehingga

pekerjaan menjadi lebih efisien dan dapat memberikan pelayanan lebih bagi konsumen, yang akhirnya dapat menghasilkan nilai tambah dan memberikan keuntungan maksimal bagi semua pihak yang berkepentingan (*stakeholder*) atas perusahaan.

2.1.4.2 Infrastruktur ERP

Menurut Wijaya dan Darudianto (2009, p22). Secara umum, infrastruktur ERP yang perlu diperhatikan adalah sebagai berikut:

- *People*

Orang yang terlibat dalam penerapan sistem ERP merupakan faktor yang sangat penting, terutama dalam hal komitmen waktu, dukungan *top management*, rasa memiliki, keterlibatan, semangat dan rasa perlawanan yang minimum.

- *Process*

Hal terpenting dalam proses yang merupakan acuan utama dalam melakukan implementasi sistem ERP adalah sebelum mengambil keputusan menggunakan sistem ERP, maka perusahaan harus sudah memiliki prosedur bisnis yang baik yang akan diterapkan dalam implementasi sistem ERP

- *Technology*

Penerapan sistem ERP identik dengan investasi yang relatif besar, dimana teknologi meliputi dari infrastruktur jaringan, *hardware*, *software*. Jaringan yang dibangun untuk

internal/*Local Area Network* dan untuk eksternal/*Wide Area Network*.

2.1.5 SAP

2.1.5.1 Sejarah SAP

Menurut Wijaya et. al.(2009, p 151). SAP merupakan salah satu sistem ERP yang populer di Indonesia. SAP didirikan sekitar 1975 di Jerman oleh 5 orang mantan karyawan yang bekerja di IBM. Singkatan SAP sebenarnya dari bahasa Jerman, yaitu *Systeme Andwendumgen Produkteinder Datenverarbeitung* atau dalam bahasa Inggris singkatan SAP adalah *Systems Applications Product in Data Processing*.

2.1.5.2 Produk – Produk SAP

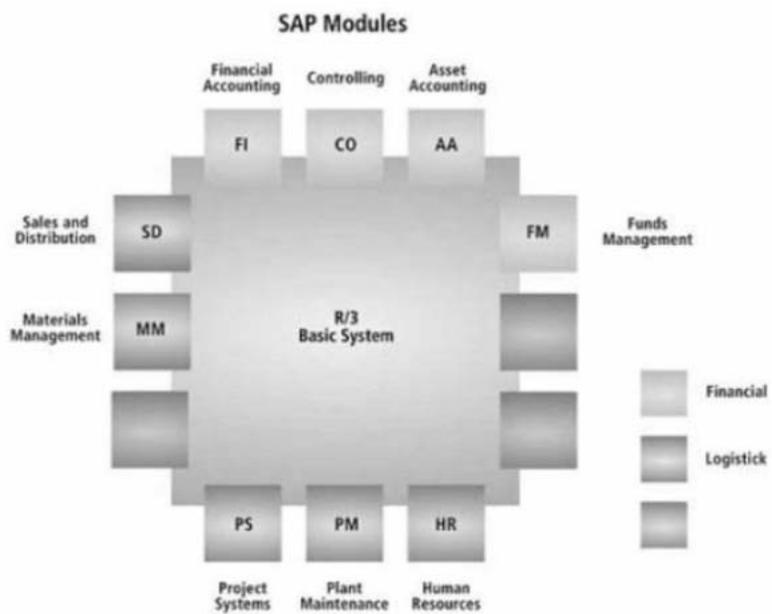
Produk yang ditawarkan oleh SAP menurut Dhewanto dan Falahah (2007, p 173) antara lain:

1. *mySAP Business Suite*: biasa digunakan untuk perusahaan yang sangat besar dan modul-modul di dalamnya sudah sangat komplit.
2. *mySAP All in One*: biasanya digunakan untuk perusahaan menengah keatas, untuk kelengkapan modulnya biasanya menyesuaikan dengan perusahaan.
3. *SAP Business One*: biasanya untuk perusahaan kecil dan modul yang ada didalamnya hanyalah berupa modul dasar saja, biasanya HR dan FI saja.

2.1.5.3 Modul – Modul SAP

Menurut Dewanto dan Falahah (2007, p172). Modul-modul dalam SAP R/3 antara lain:

1. *Financials*
 - a. *Financial Accounting (FI)*
 - b. *Controlling (CO)*
 - c. *Fixed Assets Management (AM)*
 - d. *Investment Management (IM)*
 - e. *Project System (PS)*
 - f. *Enterprise Controlling (EC)*
 - g. *Real Estate Management*
2. *Logistics*
 - a. *Sales and Distribution (SD)*
 - b. *Materials Management (MM)*
 - c. *Quality Management (QM)*
 - d. *Plant Maintenance (PM)*
 - e. *Customer Service (CS)*
 - f. *Production Planning and Control (PP)*
 - g. *SAP Retail*
3. *Human Resources*
 - a. *Personnel Management (PA)*
 - b. *Personnel Time Management (PT)*
 - c. *Payroll (PY)*
 - d. *Training and Event Management (PE)*



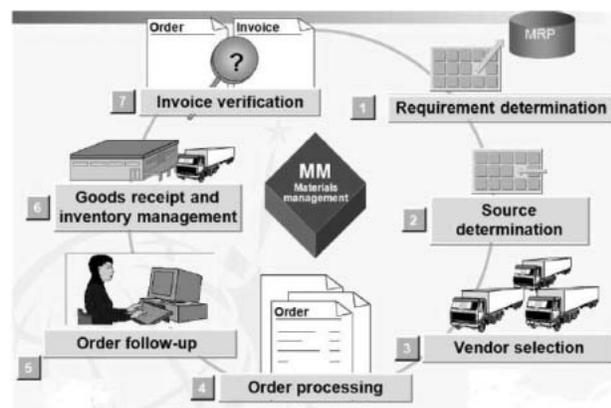
Gambar 2.1 SAP Modules

2.1.6 Procurement

Menurut Turban (2009, p234). *Procurement Management* adalah koordinasi semua aktivitas-aktivitas yang berhubungan dengan pembelian barang-barang dan jasa yang dibutuhkan untuk melengkapi misi organisasi.

2.1.6.1 Siklus Procurement

Berikut adalah siklus pada sistem *procurement* :



Gambar 2.2 Siklus Procurement

2.1.6.1.1 Requirement Determination

Pada tahap *requirement determination*, user yang ada di departemen dapat secara manual melewati tahap ini dengan membuat *purchase requisition*. Jika telah menentukan suatu prosedur MRP untuk material tersebut dalam *material master*, maka sistem SAP R/3 akan membuat *purchase requisition* secara otomatis.

2.1.6.1.2 Source Determination

Tahap *source determination* merupakan tahap penentuan calon calon *vendor* yang akan diajak untuk bekerja sama. Sebagai seorang pembeli, didukung dengan fasilitas pemilihan *source of supply*. Pembeli dapat menggunakan pembeda *source of supply* untuk membuat *request for quotation* (RFQs) dan kemudian memasukkan *quotation*. Untuk pembuatan RFQs dapat saja merujuk pada *purchase requisition* dan kondisi yang telah ada pada sistem.

2.1.6.1.3 Vendor Selection

Vendor selection merupakan tahap pemilihan *vendor* yang akan diajak kerja sama. Sistem mempermudah pemilihan dari *vendor* dengan membuat *price comparison* antara beberapa *quotation*. Secara otomatis akan mengirimkan surat penolakan

2.1.6.1.4 Order Purchasing

Pada tahap *order processing*, suatu *purchase order* (PO dapat dibuat secara manual ataupun secara otomatis oleh sistem. Pemrosesan PO dapat dilakukan pengkopian data dari dokumen lainnya seperti *purchase requisition* atau *quotation*, untuk mengurangi jumlah usaha kerja yang dilakukan. PO juga dapat dibuat berdasarkan *outline agreement*.

2.1.6.1.5 Order Follow-Up

Purchase order monitoring merupakan kegiatan memonitor status *purchase order* di dalam sistem yang dilakukan oleh pembeli. Pembeli dapat juga membedakan, contohnya apakah *delivery* atau *invoice* untuk PO tersebut telah dilakukan ataupun belum dilakukan. Dapat dilakukan peringatan kepada *vendor* tentang *outstanding deliveries*.

2.1.6.1.6 Goods Receipt and Inventory Management

Goods Receipt merupakan tahap penerimaan barang. Ketika memasukkan pengiriman yang telah datang ke dalam sistem, maka dapat dicocokkan berdasarkan PO yang relevan. Jumlah usaha kerja dapat diminimalisir dan dapat diperiksa apakah pengiriman barang, beserta jumlah barang sesuai dengan PO. Sistem akan langsung memperbarui

purchase order history dari PO dan akan menambah persediaan di gudang pada sistem.

2.1.6.1.7 Invoice Verification

Invoice verification merupakan tahap pemeriksaan *invoice*. Ketika membuat *invoice* yang berdasarkan PO yang lalu atau pengiriman yang lalu dapat diperiksa kalkulasi dan ketepatan *invoice* secara umum. Ketersediaan data dari *purchase order* dan *goods receipt* memungkinkan pembeli untuk melihat pada perbedaan kuantitas dan harga.

2.1.6.1.8 Payment Processing

Program pembayaran memungkinkan pembayaran kepada kewajiban kreditor. *Financial Accounting* mengeksekusi program ini secara reguler.

2.1.6.2 Metode – Metode Procurement

Menurut Turban (2010,p251), perusahaan – perusahaan menggunakan metode-metode yang berbeda untuk memperoleh barang dan jasa, tergantung dari apa yang mereka beli, jumlah yang dibutuhkan atau berapa banyak uang yang dipakai. Metode – metode umum *procurement* ini meliputi:

- o Membeli dari manufaktur, penjual grosir maupun pengecer dari katalog – katalog mereka dan memungkinkan negosiasi.

- Membeli dari katalog yang terhubung dengan para penjual.
- Membeli dari mal – mal industri.
- Membeli dari katalog pembeli internal dimana perusahaan menyetujui katalog – katalog *vendor* mencakup kesepakatan harga. Pendekatan ini menggunakan pengimplementasian *desktop purchasing*, dimana mengizinkan *requisition* untuk memesan secara langsung dari *vendor* dengan melewati departemen *procurement*.
- Mengadakan penawaran tender dari sistem dimana *supplier* bersaing dengan yang lainnya. Metode ini digunakan untuk pembelian dalam jumlah besar.
- Membeli dari situs pelanggan umum dimana organisasi berpartisipasi sebagai salah satu pembeli.
- Bergabung dengan suatu grup sistem pembelian dimana permintaan partisipan dikumpulkan sampai dalam jumlah yang besar, kemudian grup menegosiasikan harga/menginisiasikan sebuah proses tender.
- Berkolaborasi dengan para *supplier* untuk berbagi informasi tentang penjualan dan persediaan untuk mengurangi persediaan dan *stock – out* serta mempertinggi ketetapan waktu pengiriman.

2.1.7 Analisa dan Perancangan Sistem Informasi

2.1.7.1 System Definition

Menurut Mathiassen et al (2000, p24). *System Definition* adalah deskripsi singkat dari sistem terkomputerisasi yang diungkapkan dengan bahasa sehari-hari.

2.1.7.2 Activity Diagram

Menurut Satzinger (2005, p62). *Activity diagram* adalah suatu jenis diagram *workflow* yang mendeskripsikan aktivitas *user* dan aliran urutannya.

2.1.7.3 Class

Menurut Mathiassen et al (2000, p53). *Class* merupakan sekumpulan *object* yang memiliki struktur, *behavior pattern* dan *attribute* yang sama.

2.1.7.4 Object

Menurut Mathiassen et al (2000, p4). *Object* merupakan sebuah entitas yang memiliki identitas, *state* dan *behavior*.

2.1.7.5 Structure

Menurut Mathiassen et al (2000). *Structure* merupakan hubungan antara *class* dan *object*. *Structure* dibagi menjadi 2 yaitu *structure class* dan *structure object*.

Structure class terdiri dari 2 (dua), yaitu:

- *Generalization structure* merupakan sebuah *class* umum (*super class*) mewariskan sejumlah properti ke sekumpulan *class* anak (*sub class*).
- *Cluster structure* merupakan kumpulan dari *class* yang saling berhubungan. Pengelompokan dilakukan berdasarkan jenis, *behavior*, atau *function* yang sama.

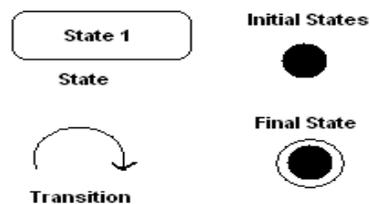
Structure object terdiri dari 2 (dua), yaitu:

- *Aggregation structure* merupakan sebuah *object* (secara keseluruhan) mencakup *object-object* bagiannya.
- *Association structure* merupakan hubungan antar *object* yang berarti.

2.1.7.6 Behavior

Menurut Mathiassen et al (2000). *Behavioral pattern* merupakan deskripsi dari kemungkinan *event-event* yang bisa muncul dalam setiap *object* di dalam sebuah *class*. Biasanya *behavioral pattern* digambarkan dengan *statechart*.

Notasi-notasi yang terdapat pada *statechart* antara lain :



Gambar 2.3 Notasi-Notasi *Statechart*

2.1.7.7 Application Domain

Menurut Mathiassen et al (2000). *Application domain* merupakan bagian organisasi yang mengatur dan memonitor, serta mengelola *problem domain*. Adapun aktivitas utama dari *application domain*, antara lain:



Gambar 2.4 Aktivitas dalam *application domain analysis*

2.1.7.8 Usage

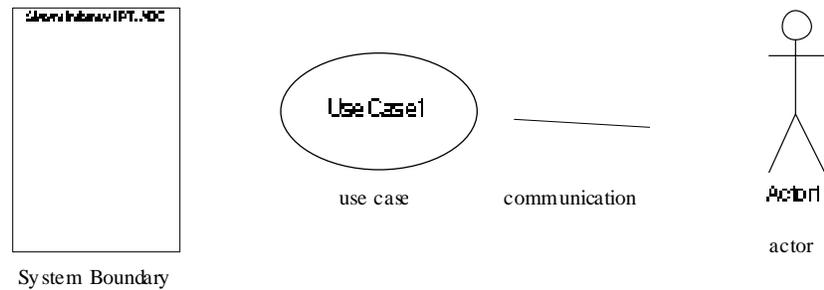
Menurut Mathiassen et al (2000). *Usage* yang menjadi prinsip di dalam *usage analysis* ini adalah menentukan *application domain* dengan menggunakan *use case*, memperkirakan perubahan sosial di dalam *application domain*, dan mengevaluasi *use case* dalam kolaborasinya dengan pemakainya.

Use case adalah suatu pola interaksi diantara sistem dengan *actor*-nya di dalam *application domain*.

Menurut Satzinger (2005, p52). *Use Case* adalah suatu aktivitas yang dilakukan, umumnya dalam merespon permintaan *user*.

Actor merupakan suatu abstraksi dari *user* atau sistem lain yang berinteraksi dengan sistem.

Use case dan *actor* dapat terlihat hubungannya di dalam sebuah diagram yang dinamakan *use case diagram*. Adapun notasi-notasi yang ada di dalamnya antara lain:



Gambar 2.5 Notasi Use Case

2.1.7.9 Function

Menurut Mathiassen et al (2000). *Function* merupakan fungsi-fungsi yang ada di dalam sebuah sistem. *Function* ini memiliki beberapa tipe, yaitu:

- *Update*, fungsi ini di aktifkan oleh sebuah *event* di *problem domain* dan menghasilkan perubahan pada keadaan *model*.
- *Read*, fungsi yang diaktifkan oleh adanya kebutuhan informasi oleh *actor*.
- *Compute*, fungsi yang di aktifkan oleh kebutuhan informasi oleh *actor*, dimana informasi itu terdiri dari perhitungan yang melibatkan informasi yang disediakan, baik oleh *actor* maupun *model*.

- *Signal*, fungsi ini diaktifkan oleh sebuah perubahan pada keadaan dan menghasilkan reaksi, yang menjadi semacam tanda bagi *actor* atau bagi sarana interfensi secara langsung di dalam *problem domain*.

2.1.7.10 Interface

Menurut Mathiassen et al (2000). *Interface* adalah sarana yang menjadikan *model* dan *function* dari sistem tersedia bagi *actor*. Ada 2 tipe interface yaitu *user interface* (*interface* yang menghubungkan pengguna dengan sistem) dan *system interface* (*interface* yang menghubungkan sistem dengan sistem lain).

Hubungan antara *interface* yang satu dengan *interface* lainnya di dalam sebuah sistem dapat digambarkan dengan menggunakan *navigation diagram*.

Ketika *usage*, *function*, dan juga *interface* sudah selesai, maka kita dapat menghubungkan ketiganya melalui sebuah diagram yang dinamakan *sequence diagram*. *Sequence diagram* ini merupakan model dinamik dari sebuah *use case* yang menampilkan interaksi antar *class* dalam periode waktu yang spesifik.

Menurut Bennet et al (2006, p252). *Sequence diagram* adalah bagian dari interaksi diagram UML, *sequence diagram* ini secara semantik sesuai dengan diagram komunikasi untuk interaksi yang sederhana.

2.1.8 Database

Menurut Connolly and Begg (2005, p15). *Database* adalah kumpulan dari data yang saling berhubungan dan penjelasan dari data tersebut dirancang untuk memenuhi kebutuhan informasi organisasi.

2.1.9 Internet

Menurut Williams dan Stacey (2005, p6). Internet adalah jaringan komputer seluruh dunia yang menghubungkan ratusan dari jutaan jaringan yang lebih kecil. Jaringan ini menghubungkan entitas edukasional, komersil, *non-profit*, militer, dan juga individu.

2.1.10 WWW (World Wide Web)

Menurut Connolly dan Begg (2005, p998). *World Wide Web* (WWW) merupakan sistem berbasis *hypermedia* yang menyediakan cara penelusuran informasi pada internet dalam cara yang tidak berurutan menggunakan *hyperlink*. Informasi pada *web* ditampilkan pada halaman *web*, yang muncul sebagai kumpulan *text*, grafik, gambar, suara dan video. Sebagai tambahan, sebuah halaman *web* dapat terdapat *hyperlink* ke halaman *web* lain, yang memungkinkan pengguna untuk berselancar dalam cara yang tidak berurutan.

2.1.11 Client/Server

Menurut Williams dan Stacey (2005, p11). *Server* atau *server* jaringan adalah komputer pusat yang menyimpan kumpulan data (*database*) dan program untuk menghubungkan atau memberikan layanan kepada PC, *Workstation* dan peralatan lainnya yang disebut *client*.

2.1.12 HTTP

Menurut Connolly dan Begg (2005, p999). *HyperText Transfer Protocol* (HTTP) menentukan bagaimana *client* dan *server* berkomunikasi. HTTP berdasarkan pada paradigma *request – response*.

Sebuah transaksi HTTP terdiri dari tahap-tahap berikut:

1. *Connection* : *client* menetapkan koneksi dengan *web server*.
2. *Request* : *client* mengirimkan pesan permintaan dengan *web server*.
3. *Response* : *web server* mengirimkan *response* kepada *client*.
4. *Close* : koneksi ditutup oleh *web server*.

2.1.13 URL (Uniform Resource Locator)

Menurut Connolly dan Begg (2005, p1002-1003). *Uniform Resource Locator* (URL) merupakan *string* dari karakter alfanumerik yang menggambarkan lokasi atau alamat dari sumber pada internet dan bagaimana sumber tersebut diakses. Sintaks dari URL cukup sederhana dan terdiri dari 3 bagian dasar: protokol yang digunakan untuk koneksi, nama *host* dan nama *path* pada *host* dimana sumber dapat ditemukan. Sebagai tambahan, URL secara *optional* dapat menentukan *port* mana yang harus dilalui koneksi (*default* adalah 80 untuk HTTP) dan *query string*, yang merupakan salah satu cara utama untuk mengirimkan data dari *client* ke *server*.

SintaksURL

:<protocol>://<host>[:<port>]/absolute_path[?arguments]

2.1.14 Online

Menurut Williams (2007). *Online* adalah menggunakan komputer atau sumber informasi lainnya yang terhubung ke jaringan untuk mengakses informasi dan layanan dari komputer lain atau sumber informasi lainnya.

2.1.15 Personal Home Page (PHP)

2.1.15.1 Pengertian PHP

Menurut *anonymous* (2011). PHP adalah bahasa *script* dengan tujuan umum yang banyak digunakan dimana sangat cocok untuk pengembangan *web* dan dapat dimasukkan dalam HTML.

Menurut Kadir (2002, p1). PHP merupakan bahasa berbentuk skrip yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya dikirim ke klien, tempat pemakai menggunakan *browser*.

Menurut Luke Welling dan Laura Thomson (2001, p2). PHP adalah *server – side scripting language* yang didesain secara spesifik untuk *web*. Dalam halaman HTML, dapat dimasukkan kode PHP yang akan dieksekusi setiap kali halaman dikunjungi. PHP kemudian diterjemahkan di *web – server* dan diubah menjadi HTML atau *output* lain yang akan dilihat oleh pengunjung halaman.

2.1.15.2 Kelebihan PHP

Menurut Sukarno (2006, p10). Kelebihan PHP adalah:

- PHP merupakan sebuah bahasa *script* yang tidak melakukan kompilasi dalam penggunaannya. Tidak seperti halnya bahasa pemrograman seperti *Visual Basic* dan sebagainya.
- PHP dapat berjalan pada *web server* yang dirilis oleh *Microsoft*, *Script IIS* atau *PWS* juga pada *Apache* yang bersifat *open source*.
- Karena sifatnya *open source*, maka perubahan dan perkembangan *interpreter* pada PHP lebih cepat dan mudah, karena banyak milis – milis dan *developer* yang siap membantu pengembangannya.
- Jika dilihat dari segi pemahaman, PHP memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.
- PHP dapat berjalan pada tiga *operating system*, yaitu *Linux*, *Unix*, dan *Windows*, dan juga dapat dijalankan secara *runtime* pada suatu *console*.

2.1.16 MySQL

2.1.16.1 Pengertian My SQL

Menurut Luke Welling and Laura Thomson (2001, p3). MySQL adalah *relational database management system* (RDBMS) yang sangat cepat dan kuat. Sebuah *database* yang memungkinkan kita untuk menyimpan, mencari, mengurutkan, dan mengambil secara efisien.

Menurut Sukarno (2006, p3). MySQL merupakan perangkat lunak untuk sistem manajemen *database(Database Management System)*. Karena sifatnya yang *open source* dan memiliki kemampuan manampung kapasitas sangat besar, maka MySQL menjadi *database* yang sangat populer di kalangan *programmer web*.

2.1.16.2 Kelebihan MySQL

Menurut Sukarno (2006, p4). Yang menyebabkan MySQL sangat populer di kalangan *web* adalah:

- My SQL tersedia di berbagai *platform*, baik itu *Linux* atau *Windows* serta dalam berbagai varian *Unix*. Misalnya MySQL yang dirilis oleh *Microsoft*, *database* ini hanya tersedia pada *platform Microsoft* saja.
- Fitur – fitur yang dimiliki MySQL sangat banyak dibutuhkan dalam aplikasi *web*, contoh: *kalusa LIMIT* yang berfungsi untuk pengaturan halaman. Atau ada pula jenis *index field FULLTEXT* yang berguna untuk *full text searching*.
- MySQL memiliki *overhead* koneksi yang rendah. Karakteristik inilah yang menjadikan SQL cocok bekerja dengan aplikasi CGI, dimana di setiap *request script* akan melakukan koneksi, mengirimkan satu atau lebih perintah SQL, lalu memutuskan koneksi lagi.

2.1.17 Add-On

Menurut *Anonymous* (2011), *Add-On* sering disebut sebagai plug-in adalah sebuah *software* yang akan membantu mengoptimalkan aplikasi perangkat lunak lain dan biasanya tidak dapat dijalankan secara independen.

2.1.18 Teknik Pengumpulan Data

2.1.18.1 Interview (Wawancara)

Menurut Sugiyono (2008, p.194), wawancara digunakan sebagai teknik pengumpulan data, apabila peneliti ingin melakukan studi pendahuluan untuk menemukan permasalahan yang ingin diteliti, dan juga apabila peneliti ingin mengetahui hal-hal dari responden yang lebih mendalam dan jumlah respondennya sedikit. Teknik pengumpulan data ini mendasarkan diri pada laporan tentang diri sendiri atau *self-report*, atau setidaknya tidaknya pada pengetahuan atau keyakinan pribadi.

2.2 Teori – Teori Khusus

2.2.1 Vendor Portal

Menurut Zabir (2008,p2). Sebuah *web portal* adalah sebuah halaman yang mengizinkan *user* untuk mengkostumisasi *home page*-nya dengan melakukan *drag* dan *dropwidjet*. Pendekatan ini memberikan kontrol penuh kepada user atas konten apa yang dilihat pada *home page*-nya, dimana halaman *web* tersebut adalah halaman yang ingin dilihat *user*, dan bagaimana *user* tersebut berinteraksi dengan konten tersebut.

2.2.2 Fit / Gap Analysis

Menurut Hoffman dan Bateson (2006, p334), *Gap Analysis* adalah suatu alat yang digunakan untuk mengetahui mengenai kondisi yang sedang berjalan di perusahaan tersebut, untuk kemudian diperbandingkan dengan sumber daya perusahaan tersebut. Hal tersebut dilakukan agar dapat mengetahui apakah suatu perusahaan sudah bergerak di proses bisnisnya secara optimal untuk memaksimalkan kinerja perusahaan tersebut.

Laporan *fit / gap* yang dihasilkan secara khusus menggambarkan perbedaan. Proses *fit / gap* mengevaluasi kebutuhan pengguna untuk proyek dan mengidentifikasi beberapa *gap* dalam fungsionalitas sekarang lawan menggunakan fungsionalitas penuh yang disediakan modul – modul perangkat lunak. *Alternatives* akan disediakan ketika *gap* dalam fungsionalitas ditemukan. Beberapa *gap* akan disetujui dengan perubahan pada proses bisnis, *custom report* atau *customization* perangkat lunak.

Gap analysis dapat dilihat melalui beberapa perspektif, yaitu :

- Organisasi (sumber daya manusia)
- Arah bisnis perusahaan
- Proses bisnis perusahaan
- Teknologi informasi

Fit / gap analysis secara khusus akan meninjau :

- Setiap proses bisnis area fungsional.
- Semua elemen data kunci yang butuh untuk diubah / diciptakan.
- Metode untuk menggunakan elemen data pasti.

Tujuan *fit / gap analysis*, yaitu untuk :

- Memastikan bahwa perangkat lunak memenuhi kebutuhan semua proses bisnis sekarang.
- Melayakkan perubahan dalam prosedur pasti.
- Mengidentifikasi tampilan penting untuk menangkap.
- Mengidentifikasi permasalahan keperluan perubahan kebijakan.

2.2.2.1 Ranking Requirement

Requirement harus diidentifikasi ke dalam tingkat prioritas mereka. Ini mengijinkan tim proyek dan sponsor tim untuk memastikan semua proses bisnis kritis yang diakomodasi selama implementasi sistem baru. Mengijinkan tim proyek untuk berfokus pada area itu yang sangat penting untuk organisasi dan juga penting ketika fungsionalitas baru dapat menambah nilai ke bisnis sekarang untuk meningkatkan proses bisnis, merealisasikan efisiensi, atau meningkatkan pelaporan. Ranking setiap kebutuhan biasanya disediakan menggunakan dokumen kebutuhan proyek. Berikut mengidentifikasi kode – kode yang digunakan dalam kolom rank laporan analisis *fit / gap* :

- H (*High* / Kebutuhan Penting), yaitu kebutuhan yang kritis, penting untuk operasi dan tanpa mereka organisasi tidak dapat berfungsi, mereka juga meliputi kebutuhan pelaporan eksternal dan internal.
- M (*Medium* / Kebutuhan Penambah Nilai), yaitu kebutuhan yang jika dipenuhi, akan meningkatkan proses bisnis secara signifikan,

kebutuhan ini biasanya kurang kritis untuk bisnis organisasi, tetapi jika dipenuhi akan memberikan keuntungan biaya signifikan pada organisasi.

- L (*Low* / Kebutuhan yang Diinginkan), adalah kebutuhan yang baik untuk dimiliki dan hanya akan menambah nilai kecil ke proses bisnis dan mungkin dipertemukan melalui *workaround* atau perubahan proses bisnis.

2.2.2.2 Tingkat Kesesuaian

Tahap selanjutnya dalam tahap analisis adalah menentukan tingkat kesesuaian di antara kebutuhan pengguna dan perangkat lunak. Berikut ini akan diuraikan kode – kode yang digunakan dalam menentukan tingkat kesesuaian untuk analisis *fit / gap* :

- F (*Fit*), artinya kebutuhan secara penuh dipenuhi oleh perangkat lunak.
- G (*Gap*), artinya perangkat lunak tidak memenuhi semua kebutuhan ini. *Comment* adalah deskripsi dari kondisi perusahaan yang sedang berlangsung.
- P (*Partial Fit*), artinya perangkat lunak memiliki fungsionalitas yang memuaskan kebutuhan *user*.

2.2.2.3 Gap Resolution

Ketika *gap* diidentifikasi dalam analisis ini, maka akan menyediakan *alternative* dan solusi yang direkomendasikan untuk memecahkan *gap*, seperti mengubah proses bisnis, merancang *workaround*, mengkustomisasi perangkat lunak, atau *interfacing* dengan sistem yang lama. Titik kunci untuk mengerti adalah kustomisasi ke

fungsionalitas perangkat lunak yang dihasilkan akan meningkatkan biaya implementasi proyek seimbang dampak *negative* pada kemampuan untuk *upgrade* ke *new release*. Hasilnya, tim proyek seharusnya mengambil pendekatan berikut ketika mencari solusi *gap* potensial. Berikut adalah penjelasan berbagai pilihan untuk resolusi *gap* :

- *Package Workaround*, pertama tim akan mengidentifikasi cara *alternative* pencapaian kebutuhan dalam proses yang dihasilkan.
- Membuat bisnis menyesuaikan *package* – Jika *package workaround* tidak mungkin, tim seharusnya merekomendasikan perubahan potensial proses bisnis untuk menyesuaikan proses dan mengeliminasi *gap*.
- *Customization* sebagai usaha terakhir, seharusnya kustomisasi diperlukan, strategi yang dipilih untuk membangun fungsionalitas eksternal baru perangkat lunak dan menampilkan *package* daripada kustomisasi *package*-nya sendiri.

2.2.3 Pengertian Sistem Informasi Pembelian

2.2.3.1 Pengertian Pembelian

Pembelian merupakan proses, cara, perbuatan membeli. Umumnya, proses pembelian pada perusahaan adalah untuk pengadaan barang atau pun bahan baku bagi keperluan perusahaan.

Pada umumnya, pembelian memiliki fungsi dalam bertanggung jawab untuk memperoleh informasi mengenai harga

barang, menentukan pemasok yang dipilih dalam pengadaan barang, dan mengeluarkan *purchase order* kepada pemasok yang dipilih.

2.2.3.2 Pengertian Sistem Pembelian

Sistem pembelian adalah suatu mekanisme bagaimana proses pembelian dapat dilakukan. Secara garis besar, transaksi yang terjadi di dalam sistem pembelian adalah sebagai berikut:

- a. Fungsi gudang mengajukan permintaan pembelian ke fungsi pembelian.
- b. Fungsi pembelian meminta penawaran harga dari berbagai pemasok.
- c. Fungsi pembelian menerima penawaran harga dari berbagai pemasok dan melakukan pemilihan pemasok.
- d. Fungsi pembelian membuat order pembelian (*purchase order*) kepada pemasok yang dipilih.
- e. Fungsi penerimaan memeriksa dan menerima barang yang dikirim oleh pemasok.
- f. Fungsi penerimaan menyerahkan barang yang diterima fungsi gudang untuk disimpan.
- g. Fungsi penerimaan melaporkan penerimaan barang kepada fungsi akuntansi.

- h. Fungsi akuntansi menerima faktur tagihan dari pemasok dan atas dasar faktur dari pemasok tersebut, fungsi akuntansi mencatat kewajiban yang timbul dari transaksi pembelian.

2.2.3.3 Sistem Informasi Pembelian

Sistem informasi pembelian merupakan sistem yang dibangun dalam mengelola segala sesuatu tentang proses pembelian, seperti mengolah bukti kas keluar, pembuatan cek, penyelenggaraan register bukti kas keluar dan register cek.

Selain itu sistem informasi pembelian juga merupakan sistem yang melayani prosedur permintaan pembelian, permintaan penawaran harga dan pemilihan pemasok, *purchase order*, dan penerimaan barang dalam sistem yang terkomputerisasi.

2.2.4 Teori Procurement

Menurut Turban (2010, p289) berpendapat bahwa *procurement* adalah keseluruhan aktivitas yang dibangun sehingga organisasi dapat memperoleh dan mendapat akses ke sumber daya(material, keahlian, kemampuan, fasilitas) yang dibutuhkan untuk menjalankan aktivitas bisnis.