

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar Sistem**

Dalam proses pengembangan suatu sistem dibutuhkan pemahaman tentang konsep-konsep dasar dari sistem informasi. Suatu sistem terdiri dari struktur dan proses. Struktur sistem merupakan unsur-unsur yang membentuk sistem tersebut, sedangkan proses sistem menjelaskan cara kerja sistem tersebut dalam mencapai tujuan tersebut. Ada dua kelompok pendekatan di dalam mendefinisikan sistem, yaitu menekankan pada prosedurnya dan pada komponen atau elemennya.

Pendekatan sistem yang menekankan pada prosedur mendefinisikan sistem sebagai berikut, “Suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama untuk melakukan suatu kegiatan atau untuk menyelesaikan sasaran tertentu. Dan pendekatan yang menekankan pada elemen mendefinisikan sistem sebagai berikut, “Suatu sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu”.

#### **2.1.2 Pengertian Sistem**

Menurut Yakub (2012:1) “Manusia hidup di dunia penuh dengan sistem, disekeliling manusia apa yang dilihat sebenarnya adalah kumpulan dari suatu sistem. Penerimaan mahasiswa baru, sistem perkuliahan, sistem perekonomian, sistem bisnis, sistem peredaran bumi, sistem transportasi dan sebagainya merupakan contoh dari sistem.”

Menurut McLeod, 2004 dalam Yakub (2012:1) “sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuua.”

Sedangkan menurut Fat dalam Hutahean (2014:1) “sistem adalah suatu himpunan suatu benda nyata atau abstrak (*a set of thing*) yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*Unity*) untuk mencapai tujuan tertentu secara efisien dan efektif.”

Berdasarkan pengertian di atas, dapat disimpulkan bahwa sistem merupakan elemen-elemen yang saling berinteraksi untuk mencapai tujuan tertentu, pendekatan sistem yang menekankan pada komponen akan lebih mudah dalam mempelajari suatu sistem harus mengerti terlebih dahulu mengenai komponen atau elemen-elemen suatu sistem dari sistem tersebut.

Pendekatan yang lebih menekankan pada elemen atau komponennya mendefinisikan suatu sistem sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu. Secara umum suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen yang terorganisir, saling berinteraksi, saling ketergantungan satu sama lain dan terpadu untuk mencapai tujuan tertentu.

### **2.1.3 Pengertian Informasi**

Menurut McLeod, (2004) dalam Yakub (2012:8) “informasi (*information*) adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya”.

Menurut Kursini dan Koniyo (2007a:7), “Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sebuah informasi”.

Menurut Hutahaean (2015c:13-14), “Sistem informasi adalah Suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung oprasi, bersifat manajerial, dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan”.

Dari definisi diatas informasi diatas dapat disimpulkan bahwa informasi adalah data mentah yang diolah menjadi data jadi yang berguna bagi semua orang yang membutuhkan data tersebut.

#### **2.1.4 Pengertian Sistem Informasi**

Menurut sutanto, (2004) dalam Puspitawati dan Anggadini (2010:14) “sistem informasi merupakan komponen-komponen dari subsistem yang saling berhubungan dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi.”

Menurut O’Brian, (2005) dalam Yakub (2012:17) “sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang menggumpulkan, mengubah, dan menyebarkan informasi dala, sebuah organisasi.”

Sedangkan menurut Samryn (2015:3) “sistem informasi yang dimaksud meliputi sekumpulan sumber daya yang dirancang sedemikian rupa untuk mengubah data menjadi informasi.”

Dari definisi diatas dapat disimpulkan bahwa sistem informasi adalah komponen-komponen dari subsistem yang saling berhubungan dalam suatu

rancangan yang penerapannya berfungsi mengubah data menjadi informasi yang berguna dalam pengambilan keputusan.

### **2.1.5 Pengertian Sistem Informasi Akuntansi**

Menurut Puspitawati dan Anggadini (2011:57) “sistem informasi akuntansi dapat pula didefinisikan sebagai suatu sistem yang berfungsi untuk mengorganisasi formulir, catatan dan laporan yang dikoordinasi untuk menghasilkan informasi keuangan yang dibutuhkan dalam pembuatan keputusan manajemen dan pimpinan perusahaan dan dapat memudahkan pengelolaan perusahaan.”

Menurut Bodnar dan Hapwood, (2006) dalam Puspitawati dan Anggadini (2011:58) “sistem informasi akuntansi merupakan sistem berbasis komputer yang dirancang untuk mentransformasi data akuntansi menjadi informasi, yang mencakup siklus pemrosesan transaksi, penggunaan teknologi informasi, dan pengembangan sistem informasi.”

Sedangkan menurut Mcleod, 2004 dalam buku Yakub (2012:75) “sistem informasi akuntansi (SIA) adalah aplikasi akuntansi perusahaan.”

Berdasarkan definisi diatas dapat disimpulkan bahwa sistem informasi akuntansi adalah sistem yang dirancang untuk mengumpulkan dan mengolah data menjadi informasi yang berguna,serta menyediakan informasi dan dapat memudahkan pengelolaan informasi oleh perusahaan dan pengambilan keputusan oleh perusahaan.

### **2.1.6 Klasifikasi Sistem**

Suatu sistem dapat diklasifikasikan dari beberapa sudut pandang. Klasifikasi sistem tersebut diantaranya; sistem tak tertentu (*probabilistik system*), sistem

abstrak (*abstract system*), sistem fisik (*physical system*), sistem tertentu (*deterministic system*), sistem tertutup (*close system*), dan sistem terbuka (*open system*).

Klasifikasi sistem menurut Yakub (2012:4) sebagai berikut :

- a. Sistem tak tentu (*probabilistic system*), adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksi dengan pasti.
- b. Sistem abstrak (*abstract system*), adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem teologia yang berisi gagasan tentang hubungan manusia dengan Tuhan merupakan contoh *abstract system*.
- c. Sistem fisik (*physical system*), adalah sistem yang ada secara fisik. Sistem computer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system*.
- d. Sistem tertentu (*deterministic system*), adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.
- e. Sistem tertutup (*close system*), sistem yang tidak bertukar materi, informasi, atau energy dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan, misalnya; reaksi kimia dalam tabung yang terisolasi.

- f. Sistem terbuka (*open system*), adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan.

### 2.1.7 Karakteristik Sistem

Untuk memahami atau mengembangkan suatu sistem, maka perlu dibedakannya unsur-unsur dari suatu sistem yang membentuknya. Menurut Fatta (2007:5) karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya:

1. Batasan (*boundary*): Penggambaran dari suatu elemen atau unsur mana yang termasuk di dalam sistem dan mana yang di luar sistem.
2. Lingkungan (*environment*): Segala sesuatu di luar sistem, lingkungan yang menyediakan asumsi, kendala, dan input terhadap suatu sistem.
3. Masukan (*input*): Sumber daya (data, bahan baku, peralatan, energy) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*output*): Sumber daya atau produk (informasi, laporan, dokumen, tampilan layer computer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*): Kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bias merupakan subsistem dari sebuah sistem.
6. Penghubung (*interface*): Tempat di mana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*): Area yang dikuasi dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energy, bahan baku, dan

sebagainya. Penyimpanan merupakan suatu media penyangga di antara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

### **2.1.8 Pengertian Penjualan**

Penjualan adalah penerimaan yang diperoleh dari pengiriman barang dagangan atau dari penyerahan pelayanan dalam bursa sebagai barang pertimbangan. Pertimbangan ini dapat dalam bentuk tunai peralatan kas atau harta lainnya. Pendapatan dapat diperoleh pada saat penjualan, karena terjadi pertukaran, harga jual dapat ditetapkan dan bebannya diketahui. Menurut Mulyadi (2008:202), Penjualan merupakan kegiatan yang dilakukan oleh penjual dalam menjual barang atau jasa dengan harapan akan memperoleh laba dari adanya transaksi-transaksi tersebut dan penjualan dapat diartikan sebagai pengalihan atau pemindahan hak kepemilikan atas barang atau jasa dari pihak penjual ke pembeli.

### **2.1.9 Penjualan Online**

Menurut O'Brien dan Marakas (2006:12), *electronic commerce* adalah “membeli dan menjual, dan pemasaran dan pelayanan dari produk, jasa, dan informasi melalui berbagai jaringan komputer”. Menurut Rainer and Turban (2009:168) *e-commerce* “merupakan suatu proses pembelian, penjualan, pengiriman, atau pertukaran produk, jasa, atau informasi melalui jaringan komputer, termasuk di dalamnya penggunaan internet”. Menurut Turban (2008: 168), “*E-commerce* adalah suatu proses membeli, menjual, transfer atau pertukaran produk, pelayanan dan informasi melalui jaringan komputer termasuk internet”

### 2.1.10 Perancangan Website

Proses pembelian atau transaksi secara *online* membuat isu keamanan bertransaksi pada situs-situs *e-commerce* menjadi isu yang sangat penting. Beberapa aspek ancaman terhadap keamanan, dari yang ringan sampai yang berat, antara lain pencurian *password* dan *username*, manipulasi data, pencurian data *credit card*. Atas dasar itulah maka diperlukan proteksi atau keamanan dalam bertransaksi ataupun berbelanja melalui situs *e-commerce*, baik untuk keperluan B2B ataupun B2C.

Ada permasalahan keamanan pada *e-commerce* merupakan permasalahan security pada web secara umum (Hendraputra, 2009:137) :

- a. *Authentication*, yaitu bagaimana proses autentikasi antara user atau customer dengan portal *e-commerce* dilakukan, sehingga hanya user atau customer yang sah yang dapat bertransaksi
- b. *Authorization*, yaitu bagaimana melakukan otorisasi, khususnya pada saat proses pembayaran, sehingga data customer benar-benar terlindungi dari hacker
- c. *Confidentiality* atau *privacy*, yaitu bagaimana melindungi data customer sehingga tidak dapat dicuri atau disadap oleh *hacker* untuk digunakan pada transaksi yang tidak sah/*valid*
- d. *Availability*, yaitu bagaimana ketersediaan sistem dari web *server* sehingga *user* dapat senantiasa melakukan transaksi yang aman, kapan saja dan darimana saja.
- e. *Non-repudiation*, yaitu nir penyangkalan, hal ini diperlukan untuk menghindari pembatalan sepihak oleh user yang telah bertransaksi.

Untuk memproteksi ataupun melindungi infrastruktur *e-commerce* dari serangan atau *hacker*, salah satunya adalah dengan menerapkan teknologi *Firewall*. Secara umum, *firewall* termasuk kedalam *network security*, karena didalamnya kita menambahkan suatu perimeter pengamanan pada konfigurasi *network* atau jaringan komputer.

Selain teknologi *firewall* juga dapat digunakan pengamanan terhadap data, dalam hal ini adalah dengan menerapkan enkripsi data, dimana data yang dikirim oleh *customer* ke web *server* portal *e-commerce* akan dienkripsi sehingga tidak dapat dibaca secara langsung oleh *hacker*.

Hendraputra (2009:139) menyatakan, “*Firewall* merupakan pemberian *priviledge access control* terhadap *network infrastructure* yang ada pada portal *e-commerce*.” Secara umum *firewall* terbagi menjadi 3 jenis, yaitu *packet filtering router*, *application level proxy / application level security*, dan *screened host filter*.

Hendraputra (2009:141) menyatakan, “*Cryptograpy* merupakan teknologi *security* yang bertujuan untuk melindungi atau memproteksi data dengan mekanisme enkripsi atau penyandian terhadap data.”

Ada 2 macam cara yang digunakan untuk mengenkripsikan data (Hendraputra, 2009:142) :

a. *Symetric Key Encryption*

Pada *symmetric key encryption*, *secret key* yang digunakan untuk mengenkripsi data di sisi web *client* akan sama dengan *secret key* yang dipergunakan untuk mendekripsi pada web *server* tujuan.

b. *Asymmetric Key Encryption*

Berbeda dengan *symmetric key encryption*, pada *asymmetric key encryption* *secret key* yang dipergunakan untuk mendekripsi di sisi web server akan berbeda dengan *secret key* yang dipergunakan di sisi *client*.

Dalam merancang sebuah situs web, peneliti perlu mengumpulkan informasi yang bertujuan untuk menjadi landasan pemikiran tentang bagaimana rancangan sebuah *website* nantinya, sehingga situsweb tersebut mampu memenuhi kebutuhan. Perancangan sebuah *website*, pada prinsipnya tidak berbeda dengan perancangan *layer* seperti pada umumnya. Oleh sebab itu rancangan *layer* yang dihasilkan harus bersifat luas, artinya teknik atau cara yang dipergunakan harus mudah dimengerti dan diadaptasi oleh setiap orang dengan lingkungan budaya yang berbeda-beda.

Untuk mendapatkan antarmuka *website* yang baik, peneliti menggunakan beberapa pedoman perancangan *website* yaitu (Kendall & Kendall, 2007 : 36):

- a. Menggunakan perangkat-perangkat professional. Menggunakan perangkat lunak yang disebut Web editor seperti *Macromedia Dreamweaver*.
- b. Memepelajari *website-website* lainnya.
- c. Gunakan sumber daya yang ditawarkan *website*
- d. Gunakan perangkat-perangkat yang telah dipelajari.

Panduan yang baik dalam merancang sebuah *website* yang baik (Kendall & Kendall, 2007 : 36-47):

- a. Gunakan perangkat desain yang professional

Perancangan harus menggunakan sebuah web editor seperti *Microsoft's FrontPage* atau *Adobe Page Mill* untuk menunjang kreatifitas dan kecepatan dalam bekerja.

b. Pelajari *website* yang sudah ada

Perhatiakn *website* lain yang kiranya berhubungan dengan apa yang hendak dikerjakan. Analisa elemen yang digunakan dan lihat bagaimana elemen itu berfungsi.

c. Perhatiakn *website* yang dikerjakan oleh desainer professional

Dengan memeperhatikan *website* para professional ini, kita bisa mengetahui bagaimana sesungguhnya pengguna berinteraksi dengan *website*, misalnya carakerja *link* pada *email*, *form* yang hendak diisi atau *chatroom*.

d. Baca buku-buku

Dengan membaca buku mengenai perancangan web, seseorang bisa menambah keahliannya.

e. Perhatikan halaman web dengan perancangan yang buruk juga

Dengan memeperhatikan rancangan yang gagal, kita bisa menghindari kesalahan yang sama.

f. Ciptakan template milik anda sendiri

Jika anda mengadopsi suatu halaman standar untuk sebagian besar halaman yang anda ciptakan, anda akan mendapatkan *website* anda berjalan dengan cepat sekali serta terlihat baik.

g. Gunakan *plug-in*, *audio* dan *video* secukupnya

Memang menyenangkan memiliki fitur-fitur dengan halaman-halaman yang terlihat profesional, tetapi jangan lupa bahwa setiap orang yang melihat *website* anda tidak memiliki setiap *plug-in* baru. Jangan mengecilkan hati pengunjung dengan halaman web anda.

#### h. Rencanakan kedepan tentang

##### 1. Struktur

Melakukan perancangan *website* adalah melakukan salah satu faktor terpenting yang harus selalu dilakukan dalam mengembangkan *website* profesional. Setiap halaman yang dihasilkan harus memiliki suatu hubungan yang jelas dengan pesan yang akan disampaikan.

##### 2. Isi

Setiap halaman harus dapat memberikan suatu informasi tertentu, yang tentunya harus dapat memberikan daya tarik tertentu sehingga para pengguna menjadi tertarik dan ingin mengunjungi *website* tersebut.

##### 3. *Text*

Penggunaan kata-kata dan kalimat yang disajikan harus dapat memberikan arti yang jelas dan mudah dimengerti.

##### 4. Grafik

Penampilan dalam bentuk grafik harus memiliki latar belakang yang mendukung dan yang disertai dengan berbagai jenis variasi tampilan, bentuk perintah dan kombinasi warna yang ditampilkan harus bisa memberikan makna yang jelas. Keluaran dalam bentuk grafik harus memperhatikan tujuan perancangan grafik jenis atau tipe data yang akan ditampilkan, siapa pemakainya, efek yang akan ditimbulkan bagi pemakai sehubungan dengan jenis grafik yang dihasilkan.

##### 5. Presentasi

Bentuk penyajian informasi harus dibuat semenarik mungkin, agar halaman-halaman informasi yang disajikan dapat memberikan suatu *image* yang baik bagi para pengguna. Tata letak penulisan, perpaduan suara yang dihasilkan, dan yang disertai dengan berbagai animasi dapat menjadi informasi awal sebelum seseorang membaca informasi dari web tersebut lebih lanjut.

#### 6. Navigasi

Adalah berhubungan dengan teknik dan cara-cara yang dipergunakan untuk melakukan pergerakan terhadap halaman-halaman yang terdapat pada perancangan *website* yang dihasilkan. Kemudahan melakukan pergerakan *mouse* atau penggunaan *toolbar* yang disediakan merupakan nilai tambah lainnya.

#### 7. Promosi

Bentuk sebuah perancangan *website* harus mengandung nilai-nilai jual, yang artinya dapat mempromosikan segala sesuatu yang akan disampaikan. Harus memiliki kemampuan untuk mengiring orang-orang dalam membuka sebuah *website*, dalam arti harus dapat menciptakan *video on demand*.

Terdapat 6 hal yang perlu diperhatikan dalam merancang sebuah *website* (Sutarman, 2008 : 264-265):

##### a. Kegunaan

Fungsi dari *website* yang akan dibangun seperti apa. Hal ini tentu akan memberikan gambaran mengenai apa yang harus dikerjakan. Misalnya untuk

menjual produk, menyediakan informasi, promosi diri, hiburan, dan lain sebagainya.

b. Tujuan

Tujuan perlu dipikirkan agar ada hal yang menjadi target dalam perancangan situs, apakah untuk mencari keuntungan *financial* ataukah untuk berbagi informasi.

c. Target Pengunjung

Siapa saja yang diharapkan untuk mengunjungi situs, sesuai dengan tujuan di atas. Dapat berdasarkan umur, lokasi, pendidikan, atau semua segmen.

d. Konten

Hal ini berkaitan dengan apa saja yang akan disajikan di situs, sesuai dengan tujuan di atas. Konten dapat mendukung tujuan dari *website* dan akan sangat mempengaruhi target pengunjung.

e. Gaya

Gaya yang akan digunakan dalam sebuah situs juga penting, dan diselaraskan dengan hal-hal di atas. Misalnya target pengunjung adalah anak-anak, maka pengaturan warna disesuaikan dengan minat anak-anak, isinya juga harus menarik bagi mereka.

f. Teknologi

Teknologi dalam hal ini menyangkut bahasa pemrograman web seperti apa saja yang digunakan, apakah hanya HTML saja ataukah ditambah dengan bahasa pemrograman lainnya seperti PHP, ASP, dan lain-lain.

Langkah-langkah untuk merancang sebuah situs web antara lain (Sutarman,2008 : 265):

a. Membuat Sketsa Desain

Perancang bias saja menuangkan ide dalam merancang tatap muka sebuah *homepage* dalam bentuk sketsa di kertas dahulu. Untuk kebanyakan orang, biasanya langkah ini dilewati dan langsung pada langkah membuat *layout* desain dengan menggunakan *software*

b. Membuat *Layout* Desain

Setelah sketsa sudah jadi, untuk memperhalus rancangan, maka digunakanlah *software* aplikasi seperti *Adobe Photoshop*, *Adobe Illustrator*, *Macromedia Firework*, dan *Macromedia Freehand*.

c. Membagi Gambar Menjadi Potongan-Potongan Kecil

Setelah *layout* desain sudah jadi, gambar-gambar tersebut dipecah menjadi potongan-potongan kecil untuk meng-*optimizewaktu download*.

d. Membuat Animasi

Animasi diperlihatkan untuk menghidupkan *homepage* agar menarik pengunjung.

e. Membuat HTML

Untuk membuat dan mengedit HTML digunakan *software HTML Editor*.

f. *Programming* dan *Script*

*Programming* untuk membuat *Websitedigunakan* agar situs web menjadi lebih interaktif. *Programming* dan *Script* ini dapat dibuat dengan menggunakan PHP, ASP, *Borland Delphi*, *Visual Basic*, *CGI*.

g. Evaluasi dan Test

Hal ini diperlukan untuk mengetahui apakah *link-link* yang ada telah berjalan dengan baik dan bukan *broken link*.

h. *Upload HTML*

Agar situs dapat diakses oleh semua orang, maka isi situs perlu di *upload* ke *web hosting*.

i. *Promosi Homepage*

Langkah terakhir adalah mempromosikan situs di *search engine* agar situs mudah dicari oleh pengunjung.

### **2.1.11 Rekayasa Web**

Menurut Pressman (2009:12-13), “rekayasa *web* memberikan sebuah *framework* yang *agile* serta ilmu untuk membangun sebuah *web application* yang berkualitas”. Hal ini terlihat cukup sederhana, tetapi sangatlah penting untuk mengerti kata kunci *agile* dan *framework*. Rekayasa Web bukanlah tugas atau aktivitas tunggal. Simarmata (2010:11), menyatakan “Rekayasa Web dihadapkan pada semua aspek-aspek dari pengembangan sistem berbasis Web, mulai dari konsep dan pengembangan untuk implementasi, evaluasi kinerja, dan pemeliharaan yang berkesinambungan”.

Sekarang ini banyak pengembang *Web* mengadopsi prinsip-prinsip rekayasa Web yang secara khusus meliputi (Simarmata, 2010:11) :

- a. Analisis dan spesifikasi kebutuhan.
- b. Teknik-teknik dan metodologi pengembangan sistem berbasis Web.
- c. Integrasi dengan sistem warisan.
- d. Migrasi dari sistem warisan ke lingkungan Web.
- e. Pengembangan aplikasi waktu nyata berbasis Web
- f. Pengujian, verifikasi, dan validasi.
- g. Penilaian kualitas, kontrol, dan jaminan.

- h. Konfigurasi dan manajemen proyek.
- i. Matriks Web untuk estimasi usaha pengembangan.
- j. Evaluasi dan spesifikasi kinerja.
- k. Pembaruan (*update*) dan pemeliharaan.
- l. Manajemen staf, tim, dan model pengembangan.
- m. Manusia dan aspek-aspek budaya.
- n. Pengembangan *user-centric*, pemodelan pengguna, dan umpan balik serta keterlibatan pengguna.
- o. Pengembangan aplikasi pengguna akhir.
- p. Pendidikan dan pelatihan.

#### **2.1.12 Navigasi Pada Website**

Menurut Mesran (2009:165) Navigasi merupakan “suatu proses dalam menunjukkan posisi *record* misalnya awal akhir dari suatu *record*”.

Menurut Sutopo (2007:245) “Struktur Navigasi adalah struktur bagaimana halaman web dihubungkan dengan halaman lain”. Menentukan struktur navigasi merupakan hal yang sebaiknya dilakukan sebelum membuat *website*.

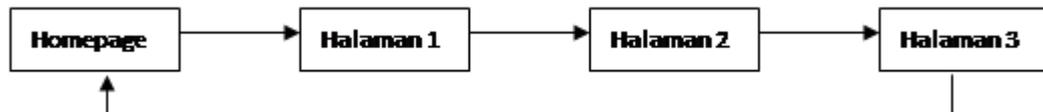
Menurut Vaughan (2006:367) navigasi adalah “memetakan struktur proyek yang dimulai pada awal tahap perencanaan untuk menggambarkan koneksi atau hubungan di antara bermacam area isi dan membantu dalam mengorganisasikan isi dan pesan”.

Menurut uraian diatas, navigasi adalah memetakan struktur suatu proses dalam menunjukkan posisi *record* untuk menggambarkan koneksi atau hubungan di antara bermacam area isi dan membantu dalam mengorganisasikan awal akhir dari suatu *record*.

Ada empat macam bentuk dasar struktur navigasi (Sutopo 2007:245) yaitu :

### 1. *Linear navigation model*

Digunakan oleh sebagian besar *website*. Informasi diberikan secara sekuensial dimulai dari satu halaman. Beberapa desainer web menggunakan satu halaman untuk masuk atau keluar dari *website*.

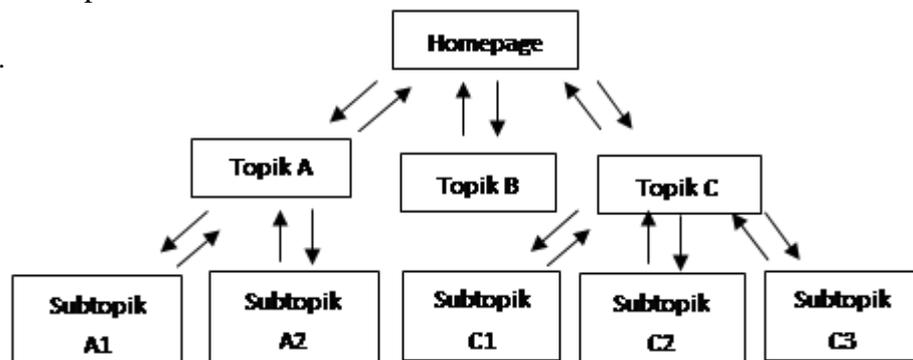


Sumber : Sutopo (2007:6)

**Gambar II.1** *Linear navigation model*

### 2. *Hierarchical model*

Diadaptasi dari *top-down design*. Konsep navigasi ini dimulai dari satu node yang menjadi *homepage*. Dari *homepage* dapat dibuat beberapa cabang ke halaman-halaman utama. Apabila diperlukan, dari tiap halaman utama dapat dikembangkan menjadi beberapa cabang lagi. Hal ini seperti struktur organisasi dalam perusahaan

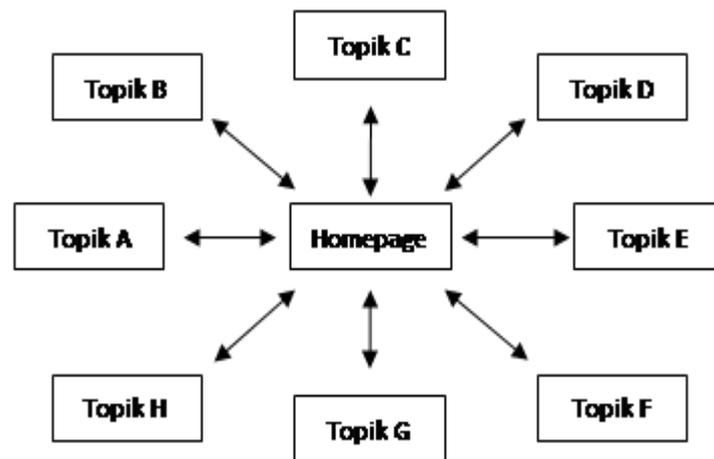


Sumber : Sutopo (2007:6)

**Gambar II.2** *Hierarchical model*

### 3. *Spoke-and-hub model*

Hanya menggunakan satu node untuk berhubungan dengan node lain. Pada model ini hanya terdapat dua macam link, yaitu dari *homepage* ke halaman tertentu, dan dari halaman tersebut kembali ke *homepage*.

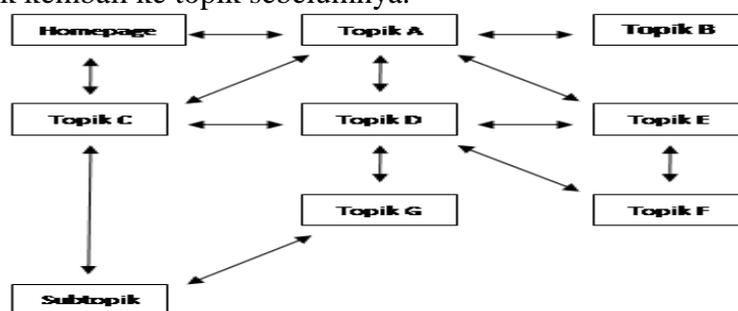


Sumber : Sutopo (2007:7)

**Gambar II.3 Spoke-and-hub model**

#### 4. Full web model

Memberikan kemampuan *hyperlink* yang banyak. *Full web model* banyak digunakan karena *user* dapat mengakses semua topik dengan subtopik dengan cepat. Namun kelemahan dari model ini, yaitu dapat berakibat *user* kehilangan cara untuk kembali ke topik sebelumnya.



Sumber : Sutopo (2007:8)

**Gambar II.4 Full web model**

## 2.2. Peralatan Pendukung (*Tools Program*)

Merupakan alat yang digunakan untuk menggambarkan bentuk logika model dari suatu sistem dengan menggunakan simbol-simbol, lambang-lambang, diagram-diagram yang menunjukkan secara tepat arti dan fungsinya. Adapun

peralatan pendukung (*tools system*) yang dijelaskan sebagai model sistem yang akan dirancang adalah sebagai berikut :

### **2.2.1 PHP**

salah satu bahasa pemrograman open source yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skrip HTML”.

Menurut Kristanto (2010:9) PHP atau kependekan dari *HypertextPreprocessor* adalah “ bahasa pemograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah *web* dan bisa digunakan pada HTML”.

Menurut Hirin dan Virgi (2011:25) PHP atau kependekan dari *HypertextPreprocessor* adalah “salah satu bahasa pemograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat di tanamkan pada sebuah skrip HTML”.

Menurut Kurniawan (2010:2) PHP (*HypertextPreprocessor*) adalah “*script* yang digunakan untuk membuat halaman *web* yang dinamis”.

Dari uraian diatas, PHP (*HypertextPreprocessor*) adalah bahasa pemograman berbentuk *script* HTML yang ditempatkan dalam *server* yang mempunyai banyak keunggulan untuk pengembangan *web*.

### **2.2.2 HTML**

Menurut Saputra (2012:1) HTML (*Hyper Text Markup Language*) adalah “bahasa yang digunakan untuk menampilkan dan mengolah tampilan pada halaman *website*”.

HTML sendiri bukan tergolong pada suatu bahasa pemrograman karena sifatnya yang hanya memberikan tanda (*marking up*) pada suatu naskah teks dan bukan sebagai program, oleh karena itu agar dapat membuat program aplikasi diatas *web*, kita terlebih dahulu harus mengenal dan menguasai HTML bilamana hendak menyisipkan setiap *script* program dalam *script* HTML.

### **2.2.3 Basis Data ( *Database* )**

Basis data atau lebih dikenal dengan *database* adalah berfungsi mirip seperti lemari, menyimpan semua data yang dimasukan oleh pengguna. Data tersebut akan disimpan dengan aman dan terkontrol.

Sedangkan Menurut Sutanta (2011:32) "Basis data (*database*) suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama – sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data, disimpan dengan cara – cara tertentu sehingga mudah digunakan atau ditampilkan kembali.

Berdasarkan penjelasan di atas penulis dapat menarik kesimpulan bahwa Basis Data adalah suatu informasi data yang diinputkan atau dimasukan kedalam suatu wadah penyimpanan berbasis komputer.

### **2.2.4 *Structure Query Language (SQL)***

Menurut Kristanto (2010:12) SQL (*Structure Query Language*) adalah "sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis".

Sedangkan menurut Rosa dan Shalahuddin (2013:46) "SQL(*Structure Query language*) adalah bahasa yang digunakan untuk mengolah data pada

RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus”.

Menurut uraian diatas, SQL(*Structure Query Language*) adalah sebuah bahasa yang dipergunakan untuk mengakses data dalam basis data *relation* untuk melakukan pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

### **2.2.5 MySQL**

Menurut Kurniawan (2010:16) Mysql adalah “salah satu jenis *databaseserver* yang sangat terkenal, termasuk jenis RDBMS (*Relational Database Management System*) yang mendukung bahasa pemograman PHP dan mempunyai *query* atau bahasa SQL yang simpel”.

Menurut Kadir (2008:348) Mysql adalah “salah satu jenis *databases server* yang sangat terkenal kepopulerannya disebabkan Mysql menggunakan SQL sebagai bahasa dasar untuk mengakses *databasenya*”.

Menurut kristanto (2010:12) MySQL adalah “sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread* dan *multi user*”.

Menurut uraian diatas, MySQL adalah sebuah program *database server* yang sangat terkenal yang menggunakan SQL sebagai bahasa dan sistem manajemen yang *multi user* untuk menerima, mengirim dan mengakses *databasenya* sangat cepat.

### **2.2.6 Unified Modelling Language (UML)**

UML (*Unifield Modeling Language*) adalah suatu metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi UML yaitu sebagai suatu bahasa yang sudah menjadi standar pada visualisasi,

perancangan dan juga pendokumentasian sistem *software*. Saat ini UML sudah menjadi bahasa standar dalam penulisan *blue print software*. Bahasa pemodelan grafis telah ada di industri perangkat lunak sejak lama. Pemicu utama di balik semuanya adalah bahwa bahasa pemrograman berada pada tingkat abstraksi yang terlalu tinggi untuk memfasilitasi diskusi tentang desain. Dengan menggunakan notasi-notasi seperti *UML*, alur logika dari perangkat lunak yang akan dikembangkan bisa mudah untuk dipahami. (Nugroho 2010:6).

Berdasarkan pendapat yang dikemukakan diatas dapat ditarik kesimpulan bahwa “*Unified Modelling Language (UML)* adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis objek (*Object Oriented Programming*)”.

### **2.2.7 Usecase Diagram**

Menurut Penulis *Usecase* diagram digunakan untuk menggambarkan secara ringkas siapa yang menggunakan sistem dan apa saja yang bisa dilakukannya. Diagram usecase tidak menjelaskan secara detail tentang penggunaan *usecase*, namun hanya memberi gambaran singkat hubungan antara usecase aktor, dan sistem.

Dan Menurut pendapat Rosa dan Shalahuddin (2013:155) “usecase diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”, ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan usecase.

### 1. Aktor

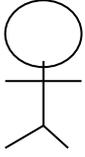
Merupakan orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

### 2. Usecase

Merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Dan berikut simbol dari diagram *usecase* Menurut Rosa dan Shalaluddin (2013:156) :

**Tabel II.1**  
**Contoh simbol diagram usecase**

| Simbol   | Deskripsi   |
|--|---|
|               | <p>Fungsionalitass yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>usecase</i></p>   |
| <p>Aktor</p>  | <p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun gambar aktor adalah gambar orang bukan berarti aktor itu orang. Biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.</p> |

|   |  |
|---|--|
| Asosiasi<br>       | Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan aktor  |
| Ekstensi<br>       | Relasi <i>usecase</i> tambahan ke sebuah <i>usecase</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek biasanya <i>usecase</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang di tambahkan. |
| Generalisasi<br> | Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>usecase</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.  |

Sumber :Rosa dan shalahuddin (2013:155)

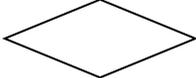
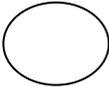
Dari pendapat diatas dapat di tarik kesimpulan bahwa usecase merupakan suatu penggambaran diagram yang berguna untuk di gunakan sebagai alir dari suatu interaksi antara aktor terhadap sistem, tapi aktor dapat berupa juga sistem.

### 2.2.8 Activity diagram

sesuai dengan namanya diagram ini menggambarkan tentang aktifitas yang terjadi pada sistem, Dari pertama sampai akhir, diagram ini menunjukkan langkah – langkah dalam proses kerja sistem yang kita buat, Menurut Satzinger et al (2010:141), “Activity diagram merupakan sebuah tipe dari diagram workflow yang menggambarkan tentang aktivitas dari pengguna ketika melakukan setiap

kegiatan dan aliran sekuensial”, sedangkan menurut rosa dan shalaluddin (2013:161)” diagram aktivitas digunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak”. Berikut contoh simbol diagram aktivitas menurut Rosa Dan Shalaluddin (2013:162)

**Tabel II.2**  
**Contoh Simbol Diagram Aktivitas**

| Simbol  | Deskripsi   |
|---|---|
| Status awal<br>    | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal        |
| Aktivitas<br>     | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja             |
| Percabangan<br>  | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu                    |
| Penggabungan<br> | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu           |
| Status akhir<br> | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir |

|   |  |
|---|--|
| <p><i>Swimlane</i></p> <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Nama Aktivitas</p> </div> | <p>Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi</p> |
|---|--|

Sumber : Rosa Dan Shalahudin (2013:162).

Dari kedua pendapat diatas ditarik kesimpulan bahwa diagram aktivitas atau activity diagram digunakan sebagai penggambaran suatu aktivitas sistem yang berjalan pada suatu bisnis atau perusahaan yang memiliki banyak aktivitas dan selalu tetap terhadap aktivitas tersebut, diagram aktivitas disini sangat penting digunakan untuk membuat suatu rancangan sistem yang baru agar dapat mengetahui aktivitas dari sistem yang akan dirancang nanti.

### **2.2.9 Sequence Diagram**

Menurut A. S. Rossa dan Shalahudin (2015d:165), “Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek”.

Menurut Santo F. Wijaya (2012d:91), “Sequence diagram merupakan suatu diagram yang menggambarkan interaksi suatu sistem dengan aktor berdasarkan user case diagram”.

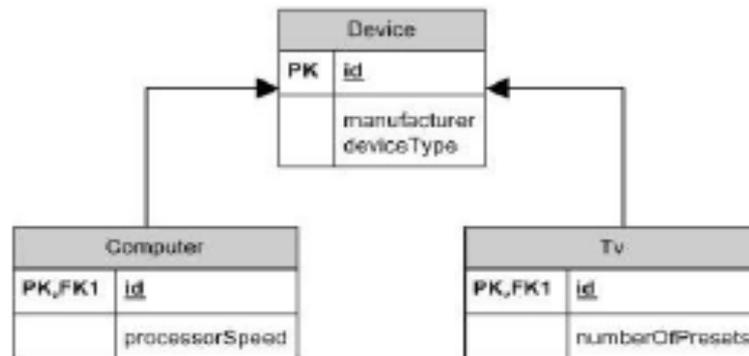
Kesimpulan dari definisi diatas Sequence diagram adalah diagram yang menggambarkan kelakuan objek pada *use case* dan menggambarkan interaksi suatu sistem dengan aktor berdasarkan user case diagram dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek.

### **2.2.10 Logical Record Structure (LRS)**

*Logical record structure* merupakan suatu struktur *record* dari tabel yang ada di *database* yang di relasikan untuk mempermudah logika dari suatu program

yang kita buat, Menurut Hasugian dan Shidiq (2012:608) memberikan batasan bahwa LRS adalah “sebuah model sistem yang digambarkan dengan sebuah *diagram-ER* akan mengikuti pola atau aturan permodelan tertentu dalam kaitanya dengan konvensi ke LRS”.

Berikut contoh penggambaran diagram LRS menurut Pradani dkk (2013:100)



Sumber : Pradani dkk (2013:100)

**Gambar II. 5 Contoh Diagram LRS**

Dari kedua pendapat diatas dapat di tarik kesimpulan bahwa Perbedaan LRS dan ERD adalah nama dan tipe *record* berada diluar *field* tipe *record* di tempatkan, LRS terdiri dari link-link diantara tipe *record*, Link ini menunjukkan arah dari satu tipe record lainnya, Banyak link dari LRS yang diberi tanda *field - field* yang kelihatan pada kedua link tipe *record*, Penggambaran LRS mulai dengan menggunakan model yang dimengerti.

### 2.2.11 Entity Relationship Diagram (ERD)

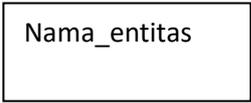
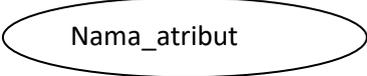
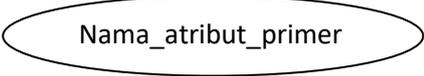
Sedangkan menurut Rosa dan Shalahuddin (2013:50) “ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka Perancangan basis data perlu menggunakan ERD”.

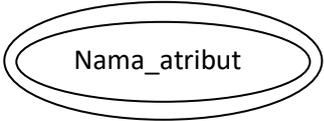
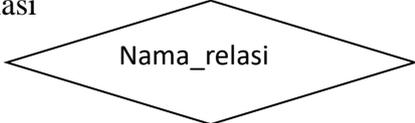
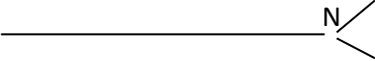
Menurut Sutanta dalam Silberschatz dkk (2011:92) “ERD (*Entity Relationship Diagram*) tersusun atas tiga komponen yaitu entitas, atribut dan kerelasian antar entitas yang secara garis besar entitas merupakan objek dasar yang terlibat dalam sistem, atribut berperan sebagai penjelas entitas dan kerelasian menunjukkan hubungan yang terjadi di antar dua entitas”.

Menurut Yakub (2008:29) ERD (*Entity Relationship Diagram*) merupakan “suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara *abstrak*”.

Menurut uraian diatas, ERD (*Entity Relationship Diagram*) adalah suatu model jaringan yang tersusun atas tiga komponen yaitu entitas, atribut dan kerelasian antar entitas yang disimpan pada sistem secara *abstrak*.

**Tabel II.3**  
**Simbol ERD**

| <b>Simbol</b>  | <b>Deskripsi</b>  |
|--|---|
| Entitas/ <i>Entity</i><br>      | Entitas merupakan data inti yang akan disimpan; bakal table pada basis data;                      |
| Atribut<br>                     | <i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas                              |
| Atribut kunci <i>primer</i><br> | <i>Field</i> atau kolom yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses |

|  |   |
|--|---|
| <p>Atribut multivalui / <i>multivalued</i></p>  | <p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p> |
| <p>Relasi</p>                                   | <p>Relasi yang menghubungkan antara entitas;</p>  |
| <p>Asosiasi / <i>association</i></p>            | <p>Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i>.</p>                   |

Sumber : Rosa dan Shalahuddin (2013:50)

### 2.2.12 *Deployment Diagram*

Menurut A. S. Rossa dan Shalahuddin (2015c:154), “*Diagram Deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”.

Menurut buku Rosa dan Shalahuddin (2014 : 154) “*Diagram deployment* atau *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi”. Diagram deployment juga dapat digunakan untuk memodelkan hal-hal berikut:

1. Sistem tambahan (*embedded system*) yang menggambarkan rancangan *device,node*, dan *hardware*.
2. Sistem *client/server*
3. Sistem terdistribusi murni
4. Rekayasa ulang aplikasi

Dari definisi diatas dapat di simpulkan bahwa *Diagram deployment* adalah konfigurasi komponen dalam proses eksekusi aplikasi