

BAB II

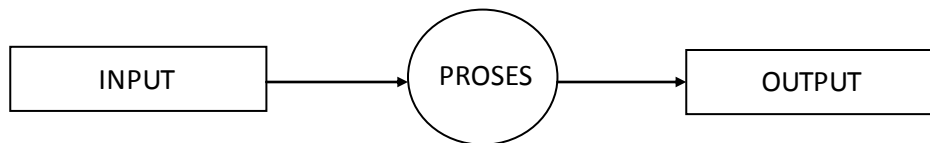
LANDASAN TEORI

2.1. Tinjauan Pustaka

A. Konsep Dasar Sistem

1. Pengertian Sistem

Menurut Taufiq (2013:2) dalam buku sistem informasi manajemen menyatakan bahwa sistem adalah kumpulan dari *sub-sub* sistem, baik yang abstrak maupun fisik yang saling berintegrasi dan berkolaborasi untuk mencapai suatu tujuan tertentu. Gambaran umum model dari sebuah sistem sebagai berikut:



Sumber: Taufiq (2013:3)

Gambar II. 1 Model Umum Sistem

2. Karakteristik Sistem

Menurut Taufiq (2013:4) Sistem yang baik adalah sistem yang mempunyai tujuan, batasan, subsistem, *input*, *proses*, *output* dan *feedback* (umpan balik) secara jelas. Untuk lebih memahami, penjelasan ciri-ciri sistem yang terintegrasi dengan sistem atau subsistem itu sendiri. Berikut adalah karakteristik dari suatu sistem yang membedakan dengan sistem lainya, diantaranya yaitu:

a. Tujuan Sistem

Sistem yang baik adalah sistem yang memiliki tujuan yang jelas dan terukur yang memungkinkan untuk dicapai dan memiliki langkah-langkah yang terstruktur untuk mencapainya.

b. Sistem (*Boundary*)

Batas sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau dengan lingkungan luarnya.

c. Subsistem

Subsistem merupakan komponen-komponen atau sistem-sistem yang lebih kecil ada didalam sistem itu sendiri.

d. Masukan Sistem (*Input*)

Adalah merupakan semua kegiatan (pencatatan, pengetikan, pengeditan) atau memasukan data fisik ataupun non fisik (*logic*).

e. Keluaran Sistem (*Output*)

Adalah hasil dari masukan sistem yang diproses, *output* sering juga disebut informasi yang bermanfaat bagi penggunaanya.

f. Proses

Proses adalah suatu kegiatan yang merubah *input* sehingga menjadi *output* yang memiliki nilai tambah atau lebih berguna lagi, yang dimaksud adalah data menjadi informasi.

g. Umpan Balik (*Feedback*)

Merupakan sebuah kegiatan yang memasukan *output* kembali kedalam input, dalam hal ini *output* sebagai informasi dan *input* adalah data.

B. Konsep Dasar Informasi

1. Pengertian Informasi

Menurut Taufiq (2013:15) dalam buku sistem informasi manajemen menyatakan bahwa informasi adalah data-data yang diolah sehingga memiliki nilai tambah dan bermanfaat bagi pengguna. Sedangkan data merupakan representasi dari fakta atau gambar mengenai suatu objek atau kejadian.

2. Kualitas Informasi

Sebuah informasi yang berkualitas adalah informasi yang secara umum dapat dikatakan memenuhi apa yang dibutuhkan oleh pengguna. Ciri-ciri informasi yang berkualitas menurut Raymond McLeod dalam buku Menurut Taufiq (2013:15) menyatakan bahwa sebagai berikut:

a. Informasi Harus Akurasi (*Accurate*)

Akurat berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias, artinya akurasi merupakan tingkat keakuratan sebuah informasi.

b. Tepat Pada Waktu (*Timelines*)

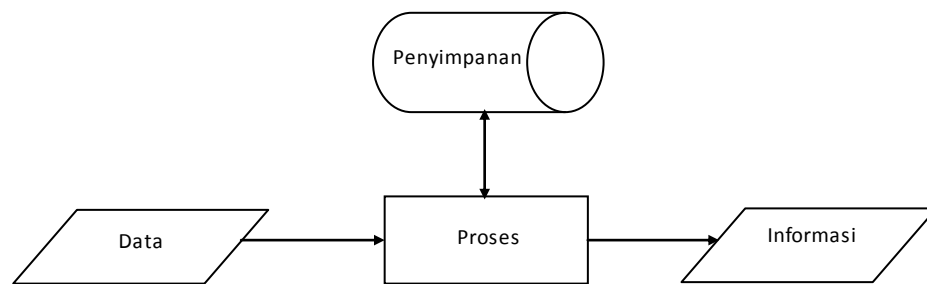
Informasi yang dihasilkan dari sistem sebisa mungkin dapat disajikan saat itu juga, karena informasi yang kita butuhkan merupakan acuan dalam pengambilan keputusan.

c. Relevan (*Relevance*)

Informasi harus sesuai dengan kebutuhan yang berhubungan dengan masalah yang dihadapi.

d. Kelengkapan

Kelengkapan informasi dapat ditunjukkan dari menjawab informasi tersebut terhadap pertanyaan atau kebutuhan pengguna.



Sumber: Taufiq (2013:15)

Gambar II. 2 Transformasi Data Menjadi Informasi

C. Konsep Dasar Sistem Informasi

1. Pengertian Sistem Informasi

Menurut Taufiq (2013:17) dalam buku sistem informasi manajemen menyatakan bahwa Sistem Informasi adalah kumpulan dari *sub-sub* sistem yang saling berintegrasi dan berkolaborasi untuk menyelesaikan masalah tertentu dengan cara mengolah data menggunakan komputer sehingga memiliki nilai tambah dan bermanfaat bagi pengguna.

Adapun komponen-komponen sistem informasi berbasis komputer menurut James A O'Brien dalam Taufiq (2013:19) adalah Manusia, *Hardware*, *Software*, Data, dan Jaringan. Berikut penjelasan dari kelima komponen tersebut diantaranya:

a. Manusia

Merupakan meliputi semua orang yang melakukan aktivitas dalam sistem baik sebagai pengguna maupun sebagai pengembang *system* informasi.

b. *Hardware*

Perangkat keras adalah alat yang digunakan untuk menerima inputan.

c. *Software*

Perangkat lunak adalah kumpulan intruksi (program) elektronik yang digunakan menyuruh *computer* untuk melakukan sesuatu sehingga menghasilkan *output* yang diinginkan

d. *Data*

Merupakan bahan mentah yang perlu diolah sehingga menghasilkan informasi atau keterangan, baik kualitatif maupun kuantitatif yang menunjukkan fakta.

e. *Jaringan*

Jaringan adalah sistem *computer*, telepon, atau piranti komunikasi lain yang terinterkoneksi sehingga mampu saling berkomunikasi dan juga dapat bertukar data atau informasi.

D. Konsep Dasar *Website*

1. *Internet*

Interconnected network atau yang lebih dikenal dengan *internet* merupakan hasil dari perkembangan teknologi. Menurut JE (2010:2) memberikan definisi bahwa “*Internet* adalah sebuah sistem jaringan yang menghubungkan berbagai komputer dari belahan dunia untuk saling terhubung dan bertukar data serta bertukar informasi”. Dengan adanya *internet* maka sebuah komputer akan dapat berhubungan dengan komputer lain dimana pun berada namun harus memiliki fasilitas *internet* sehingga kita dapat bertukar informasi, seperti bertukar

file, berbincang-bincang dengan mengirimkan *e-mail (Electronic Mail)* ataupun dengan IRC (*Internet Relay Chat*). Penggunaan *internet* memungkinkan kita untuk mendapatkan informasi dari komputer yang ada didalam kelompok tersebut.

2. Website

Perkembangan *internet* yang pesat telah membuat sebuah dunia baru, yang biasa kita kenal sebagai dunia maya. *World Wide Web (www)* atau lebih dikenal dengan *website*, ditinjau dari segi historisnya, *website* merupakan salah satu layanan yang ditawarkan oleh *internet* itu sendiri. Bermula dikembangkan oleh Tim Berners-Lee ketika bekerja di CERN. Untuk membaca atau melihat sistem dari *www* atau *website* itu sendiri menggunakan *tools* yang dikenal dengan istilah *browser*.

Menurut Hidayat (2010:2) dalam bukunya pengertian *website* adalah Kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Beberapa orang memandang bahwa *world wide web (www)* atau *website* merupakan layanan *internet* yang paling mudah digunakan. Berikut beberapa istilah yang berkaitan dengan *website*:

a. Homepage

Homepage menurut Yuhefizar (2008:161) adalah “halaman awal sebuah *website* yang diakses berdasarkan *domainnya*”.

b. Hyperlink

Hyperlink menurut Retnoningsih (2007:2) adalah “teks yang menghubungkan halaman-halaman *web* lainnya”. Pengguna *website* akan mengetahui apabila sedang berada pada *hyperlink* jika kursor berubah menjadi simbol tangan.

c. HTTP

HTTP atau *HyperText Transfer Protocol* menurut Priyatno (2009:18) yaitu “jalur yang digunakan untuk mentransfer data didalam jaringan *internet*”.

d. Web Browser

Web Browser menurut Limantara (2009:1) adalah “aplikasi perangkat lunak yang memungkinkan penggunanya untuk berinteraksi dengan teks, *image*, video, *games* dan informasi lainnya yang berlokasi pada halaman web pada *world wide web* (www) atau *Local Area Network* (LAN)”. Dengan *web browser*, seorang pengguna bisa mengakses informasi yang disediakan pada banyak *website* secara cepat dan mudah.

e. Domain

Domain menurut Hidayat (2010:9) adalah “alamat unik didunia *internet* yang digunakan untuk mengidentifikasi sebuah *website*, atau dengan kata lain *domain* adalah alamat yang digunakan untuk mencari dan menemukan sebuah *website* pada dunia *internet*”. Nama *domain* membentuk suatu identitas tunggal untuk sekumpulan komputer yang digunakan oleh suatu perusahaan.

f. Uniform Resource Locator

Uniform Resource Locator atau URL menurut Suyanto (2007:222) adalah “alamat sebuah *file* yang dapat diakses di *internet*”. URL dapat digunakan untuk membantu komputer untuk menemukan lokasi pasti halaman web pada server. URL berisi nama protokol *internet* yang

dibutuhkan untuk mengakses *file*, nama komputer yang berisi *file* tersebut dan *direktori* dimana *file* tersebut berada.

g. *Web Server*

Ketika melakukan *browsing internet*, *browser* pengguna bertindak sebagai klien yang akan disambungkan ke berbagai *web server* dan *web server* akan kembali menampilkan informasi dari suatu situs. Definisi *web server* menurut Saputra dalam bukunya (2010:2) adalah "Server *internet* yang digunakan sebagai koneksi dan *transfer* data (HTML, asp, aspx, php, js, dan lain sebagainya)". Pada *windows*, *web server* defaultnya adalah IIS. Namun pada saat ini *web server* yang lebih banyak digunakan adalah *apache* karena *web server* ini merupakan *web server* yang paling kompatibel dan andal dengan PHP dan MySQL.

3. *File Transfer Protocol*

File Transfer Protocol atau FTP menurut Madcoms (2010:18) merupakan "suatu layanan *internet* yang mentransfer *file-file* dari satu komputer ke komputer lain". Layanan ini bekerja dengan memanfaatkan suatu komputer yang berfungsi sebagai pusat penyimpanan data yang disebut sebagai *server* FTP. Ada dua istilah yang lazim yang digunakan berkaitan dengan *transfer file* adalah sebagai berikut:

a. *Download*

Download menurut Priyatno (2009:92) adalah "istilah yang digunakan saat proses *transfer* data dari komputer server ke komputer lokal". *File* hasil *download* akan tersimpan didalam media penyimpanan di komputer lokal atau komputer yang digunakan oleh pengguna.

b. *Upload*

Upload menurut Priyatno (2009:92) adalah “proses men-*transfer* data dari komputer lokal ke komputer *server*”. *File* hasil *upload* akan tersimpan di komputer *server* dengan mendapatkan *hosting* dari penyedia layanan *hosting*.

E. Pengenalan HTML

HTML merupakan singkatan dari *Hypertext Markup Language*. Menurut Hidayatullah dan Kawistara (2014:13) mengemukakan bahwa “HTML adalah bahasa *standard* yang digunakan untuk menampilkan halaman *web*”.

HTML mirip dengan dokumen teks biasa, hanya dalam dokumen ini , tulisan mengandung beberapa instruksi yang ditandai dengan kode tertentu yang lebih dikenal dengan TAG tertentu.

Secara garis besar, terdapat 4 jenis elemen dari HTML menurut Winarno, dkk (2013:01), yaitu:

1. Struktural

Tanda yang menentukan level atau tingkatan dari tulisan (misalnya, `<h1>Golf</h1>` memerintahkan browser menampilkan “Golf” sebagai tulisan tebal besar yang menunjukkan sebagai Heading 1).

2. Presentasi

Tanda yang menentukan tampilan dari sebuah tulisan tidak peduli dengan level dari tulisan tersebut (contoh, `<b>tebal</b>` akan menampilkan tulisan yang tebal).

3. *Hiperteks*

Tanda yang berfungsi membuat *link* di dokumen, baik *link* di dokumen itu sendiri atau ke dokumen lain (contoh, `<a href="bbb.com">bbb</a>`).

4. Elemen Widget yang membuat objek-objek lain seperti tombol (`<button>`), list(`<li>`), dan garis horizontal (`<hr>`).

F. Pengenalan PHP

Menurut Winarno, dkk (2013:59) mengemukakan bahwa "PHP adalah bahasa pemrograman yang memungkinkan anda menggenerate kode HTML secara dinamis, artinya anda bisa membuat tampilan halaman *web* yang dinamis, bisa berubah-ubah sesuai dengan keinginan programmernya".

PHP adalah bahasa *script* yang sangat cocok untuk pengembangan *web* dan dapat dimasukan ke dalam HTML. Pada suatu halaman *web* dinamis, PHP berfungsi sebagai bahasa pemrograman yang menjalankan suatu perintah tertentu, sedangkan HTML berfungsi sebagai struktur dari desain halaman *web*. Ketika sebuah halaman *web* dinamis dibuka pada *web* browser, pertama kali yang terjadi adalah server memproses semua perintah PHP yang ada, kemudian menampilkan hasilnya pada format HTML ke *web browser*, sehingga yang ditampilkan ke *web browser* hanya tampilan desain dari HTML saja, sedangkan skrip PHP bekerja dibelakang layar.

PHP merupakan *software opensource*, jadi PHP bisa digunakan oleh siapa saja secara bebas tanpa harus membayar, dengan kata lain PHP adalah gratis. PHP juga merupakan *software cross platform*, jadi bisa berjalan dengan baik pada windows maupun unix (linux).

Skrip PHP berkedudukan sebagai tag dalam bahasa HTML. Sebagaimana diketahui, HTML (*Hypertext Markup Language*) adalah bahasa standar untuk membuat halaman-halaman *web*. Kode *Personal Home Page Tools* diawali dengan `<?PHP` dan diakhiri dengan `?>`. Pasangan kedua kode inilah yang berfungsi sebagai *tag* kode PHP. Berdasarkan tag inilah, pihak *server* dapat memahami kode PHP dan kemudian memprosesnya. Hasilnya dikirim ke *browser*.

Model kerja PHP (*Personal Home Page*) pada prinsipnya serupa dengan kode HTML (*Hypertext Markup Language*). Hanya saja, ketika berkas PHP (*Personal Home Page*) yang diminta didapatkan oleh *web server*, isinya segera dikirimkan ke mesin PHP (*Personal Home Page*) dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke *web server*, *web server* menyampaikan ke *client*.

Adapun kelebihan-kelebihan dari PHP yaitu:

- a. PHP merupakan sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya. Tidak seperti halnya bahasa pemrograman aplikasi yang lainnya.
- b. PHP dapat berjalan pada *web server* yang dirilis oleh Microsoft, seperti IIS atau PWS juga pada apache yang bersifat *open source*.
- c. Karena sifatnya yang *open source*, maka perubahan dan perkembangan interpreter pada PHP lebih cepat dan mudah, karena banyak *milis-milis* dan *developer* yang siap membantu pengembangannya.
- d. Jika dilihat dari segi pemahaman, PHP memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.

- e. PHP dapat berjalan pada 3 *operating sistem*, yaitu: Linux, unix, dan windows, dan juga dapat dijalankan secara *runtime* pada suatu *console*.
- f. PHP bersifat bebas dipakai (free).

Kode PHP diawali dengan tag `<?php` dan diakhiri dengan tag `?>`. Apabila kita melakukan konfigurasi terhadap PHP ini untuk mengizinkan menggunakan tag pendek (*short tag*) dengan mengubah nilai `short_open_tag` menjadi *on*, maka tag tersebut dapat diganti dengan `<?` dan `?>`. Dalam PHP 5, nilai default dari `short_open_tag` adalah *off*. Selain itu PHP kita juga dapat menggunakan tag gaya ASP, `<%` dan `%>`, dengan mengubah nilai `asp_tags` dalam file `php ini` menjadi *on*.

Perintah `echo` didalam PHP berguna untuk mencetak nilai, baik teks maupun numerik, kelayar *webbrowser*. Selain `echo` kita juga dapat menggunakan perintah `print`.

Setiap perintah atau statemen didalam kode PHP harus diakhiri dengan tanda titik koma atau semicolon(`;`).

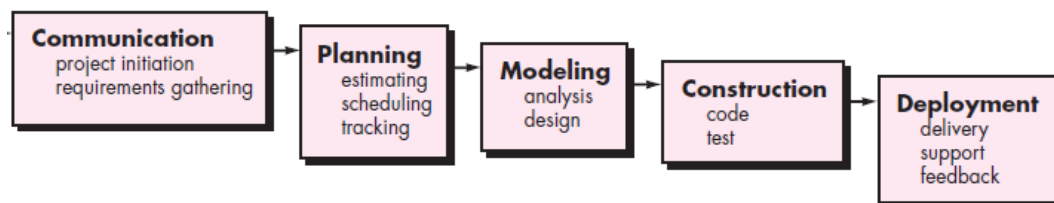
Hasil yang akan diberikan oleh kedua kode diatas adalah sama. Meskipun, demikian, penggunaan `echo` lebih banyak digemari oleh para programmer *web*. Selain itu didalam sintaks PHP kita juga dapat menambahkan komentar. Komentar berfungsi untuk memberikan informasi kepada orang yang membaca kode program yang kita buat, tapi baris-baris komentar tersebut akan diabaikan oleh PHP.

G. Model Waterfall

Metode ini merupakan metode yang sering digunakan oleh penganalisa sistem pada umumnya.

“Model *waterfall* sering disebut siklus hidup klasik, menunjukkan sistematis, pendekatan yang sekuensial untuk pengembangan perangkat lunak yang diawali dengan spesifikasi pelanggan persyaratan dan berkembang melalui perencanaan, pemodelan, konstruksi, dan penyebaran, yang berpuncak pada dukungan yang berkelanjutan dari perangkat lunak yang lengkap (Pressman, 2010:39).

Adapun langkah-langkah Model *waterfall* menurut Pressman (2010:39) adalah :



Sumber: Pressman (2010:39)

Gambar II. 3 Model Waterfall

a. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan *software*, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan pihak sekolah/madrasah, maupun mengumpulkan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

b. *Planning*

Proses *planning* merupakan lanjutan dari proses *communication* (*analysis requirement*). Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang akan dilakukan.

c. *Modeling*

Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada rancangan struktur data. Arsitektur *software*, representasi *interface*, dan detail (algoritma) procedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement*.

d. *Construction*

Construction merupakan proses membuat kode. *Coding* atau pengkodean merupakan penerjemahan desain dalam bahasa yang bisa di kenali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk demikian bisa diperbaiki.

e. *Deployment*

Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

H. *Unified Modelling Language*(UML)

“UML (*Unified Modeling Language*) adalah bahasa pemodelan dalam sistem atau perangkat lunak yang berparadigma berorientasi objek” (Nugroho, 2010:06).

Menurut Rosa dan Shalahuddin (2013:137) “*Unified modeling language* (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

Hal ini disebabkan karena UML menyediakan bahasa pemodelan *visual* yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain.

UML mendefinisikan berbagai diagram dalam pembuatan sistem informasi akademik Madrasah Aliyah Negeri (MAN) Rengasdengklok tidak menggunakan *class* maka diagram UML yang digunakan diantaranya adalah: *use case diagrams*, *activity diagrams*, *sequence diagrams* dan *deployment diagrams*.

1. *Use Case Diagram*

Use Case secara defitif sesungguhnya merupakan “urutan aksi-aksi (sering juga disebut sebagai *flow-of-events*) yang akan dilakukan sistem atau perangkat lunak untuk memberikan hasil atau nilai tertentu pada masing-masing *actor*” (Nugroho, 2010:86).

Menurut Whitten dalam Widodo (2011:21) mengartikan bahwa “*Use Case* sebagai urutan langkah-langkah yang secara tindakan saling terkait (skenario),

baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal”.

Use case bekerja dengan cara mendeskripsikan tipikal antara pengguna sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut *scenario*. Setiap *scenario* mendeskripsikan urutan kejadian. Setiap urutan diinisialisasi oleh orang, sistem yang lain, perangkat keras atau urutan waktu. Dengan demikian secara singkat bisa dikatakan.

“*Use Case Diagram* adalah diagram fungsional dalam arti bahwa mereka menggambarkan fungsi dasar dari suatu sistem, yaitu apa yang dapat dilakukan pengguna dan bagaimana sistem harus menanggapi tindakan pengguna” (Dennis, dkk 2009:166).

2. Activity Diagram

“*Activity Diagram* digunakan untuk model perilaku dalam independen proses bisnis Suatu objek. Dalam banyak hal, *Activity Diagram* dapat dipandang sebagai diagram aliran data yang canggih yang dapat digunakan dalam hubungannya dengan analisis terstruktur” (Dennis, dkk. 2009:159)

“*Activity Diagram* atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak” (Rosa dan Shalahuddin, 2013:161). *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. *Activity diagram* tidak menunjukkan apa yang terjadi, tetap tidak menunjukkan siapa yang melakukan apa.

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.

3. Component diagram

Menurut Rosa dan Shalahuddin (2013:148), menjelaskan bahwa “*Component Diagram* atau diagram komponen dibuat untuk menunjukan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem”. Komponen bisa berupa *table*, *file*, *file exe* dan lain-lain. Suatu komponen bisa mengakses *service-service* yang ada di komponen lain dengan cara *import interface*. Sedangkan komponen yang menyediakan *service* menggunakan *export interface*.

Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. Diagram komponen menurut Rosa dan Shalahuddin (2013:148) juga dapat digunakan untuk memodelkan hal-hal berikut :

- a. *Source code* program perangkat lunak
- b. Komponen *executable* yang dilepas ke *user*
- c. Basis data secara fisik
- d. Sistem yang harus beradaptasi dengan sistem lain
- e. *Framework sistem*, *framework* pada perangkat lunak merupakan kerangka kerja yang dibuat untuk memudahkan pengembang dan pemeliharaan aplikasi.

4. Deployment diagram

Menurut Rosa dan Shalahuddin (2013:154) “*Deployment diagram* atau diagram *deployment* menunjukan konfigurasi komponen dalam proses eksekusi

aplikasi”. Bagian *hardware* adalah node yaitu nama untuk semua jenis sumber komputasi. Ada dua tipe node yaitu *processor* dan *device*. *Processor* adalah node yang bisa mengeksekusi sebuah komponen sedangkan *device* tidak. *Device* adalah perangkat keras seperti printer, monitor dan perangkat keras lainnya.

Deployment Diagram menyediakan gambaran bagaimana sistem secara fisik akan terlihat. Sistem terdiri dari node-node dimana setiap node diwakili untuk sebuah kubus. Garis yang menghubungkan antara dua kubus menunjukkan hubungan diantara kedua node tersebut.

I. Entity Relation Diagram (ERD)

Entity Relation Diagram (ERD) atau yang dikenal juga dengan *Diagram Entity-Relationship* (Diagram E-R) merupakan suatu model yang menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

Menurut Fatansyah (2007:79) mengemukakan bahwa “*Model Entity-Relationship* yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari dunia nyata yang kita tinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan *Diagram Entity-Relationship*”.

“*Entity Relation Diagram* adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antar entitas” (Simarmata dan Paryudi, 2010:67).

Notasi-notasi simbolik di dalam *Entity Relation Diagram* yang dapat digunakan adalah:

1. Entitas

Digambarkan dengan kotak persegi panjang dan digunakan untuk menunjukkan sekumpulan orang, tempat, objek atau konsep dan sebagainya yang menunjukkan dimana data dicatat atau disimpan.

2. Hubungan atau Relasi

Digambarkan dengan kotak berbentuk *diamond* atau belah ketupat dengan garis yang menghubungkan ke entitas yang terkait. Maka *relationship* diberi nama dengan kata kerja. Hubungan atau relasi menunjukkan abstraksi dari sekumpulan hubungan yang mengaitkan antara entitas yang berbeda.

3. Atribut

Digambarkan dengan bentuk elips. Atribut menunjukkan karakteristik dari tiap entitas atau sesuatu yang menjelaskan entitas atau hubungan. Sehingga atribut dikatakan elemn dari entitas dan relasi. Dari setiap atribut entitas terdapat satu atribut yang dijadikan sebagai kunci (*key*). Beberapa jeni kunci tersebut antara lain : *Primary key*, *Candidate key*, *Composite key*, *Secondary key*, *Alternate key* dan *Foreign key*.

4. Tingkat Hubungan (*Cardinality*)

Entity Relation Diagram (ERD) juga menunjukkan tingkat hubungan yang terjadi, dilihat dari segi kejadian atau banyak tiidaknya hubungan antara entitas tersebut, diantaranya ada tiga kemungkinan hubungan yang ada, yaitu:

a. Satu ke satu (*one to one* (1:1))

Tingkat hubungan dinyatakan satu ke satu jika suatu kejadian pada entitas pertama hanya mempunyai satu hubungan dengan satu kejadian maupun satu hubungan dengan satu kejadian pada entitas kedua. Demikian juga

sebaliknya, satu kejadian pada entitas yang kedua hanya bisa mempunyai satu hubungann dengan satu kejadian pada entitas yang pertama.

b. Satu ke banyak (*one to many* (1:M))

Tingkat hubungan satu ke banyak (1:M) adalah sama dengan banyak ke satu (M:1), tergantung dari arah mana hubungan-hubungan tersebut dilihat.

Untuk satu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian yang kedua, sebaliknya banyak kejadian pada entitas yang kedua hanya bisa mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.

c. Banyak ke banyak (*many to many* (M:N))

Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai hubungan dengan kejadian pada entitas lainnya, baik dilihat dari entitas yang pertama maupun dilihat dari entitas yang kedua.

J. Logical Relational Structures (LRS) atau Model Relasional

Menurut Kusrini (2007:18), “Model relasional adalah kumpulan tabel-tabel untuk merepresentasikan data dan relasi antar data – data tersebut”. Setiap tabel terdiri atas kolom-kolom dan setiap kolom mempunyai nama yang unik.

Logical Relational Structures (LRS) dibentuk dengan nomor tipe *record*. Beberapa tipe record digambarkan oleh kotak empat persegi panjang dan dengan nama yang unik.

K. Pengujian Web metode *Black Box Testing*

Pengujian *web* diperlukan untuk penilaian dari keberhasilan program pembuatan *web* agar dapat segera diketahui kelemahan atau fungsi dari *web* yang dibuat. Menurut Simarmata (2010:316), “*Black Box Testing* adalah pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan”.

Black box testing atau disebut juga dengan *functional testing* berfokus pada perilaku *eksternal* dari suatu perangkat lunak atau berbagai komponennya sambil melihat objek yang akan di uji sebagai sebuah kotak hitam untuk proteksi dan juga memverifikasi fungsi-fungsi *eksternal* yang disediakan oleh perangkat lunak. Pengujian *functional* meliputi seberapa baik sistem melaksanakan fungsinya termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar dan integrasi.

2.2. Tinjauan Jurnal (*Related Research*)

Dalam proses pengolahan data dan penyediaan informasi akademik sekolah perlu digunakannya suatu alat bantu dalam hal ini berupa aplikasi komputerisasi yang dapat mempermudah dan mempercepat sistem informasi akademik pada Madrasah Aliyah Negeri (MAN) Rengasdengklok.

Berikut beberapa tinjauan penelitian terdahulu (jurnal) yang dapat memperkuat alasan dibuatnya sistem informasi akademik pada Madrasah Aliyah Negeri (MAN) Rengasdengklok, diantaranya:

Menurut Dengan dan Marisa (2009:29), Sistem informasi akademik berbasis *web* ini dirancang untuk sebagai solusi untuk mengelola bagian akademik dalam penyajian laporan nilai serta keaktifan siswa dan sistem informasi ini bersifat *intern*.

Menurut Riyadi, dkk dalam jurnalnya (ISSN : 2302-7339 Vol. 09 No. 40 2012) dengan dibangunnya sistem informasi berbasis *website* diharapkan dapat memberikan kemudahan dalam aktivitas-aktivitas akademik khususnya seperti proses pencarian data guru yang dibutuhkan oleh siswa atau orang tua/ wali tidak perlu lagi mengantri di tata usaha dan pencarian materi ajar dapat dilakukan kapan saja.