

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

2.1.1. Konsep Dasar Model Pengembangan Sistem

A. Pengertian Sistem

Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai berikut:

Menurut Fitzgrald dalam Puspitawati dan Anggadini (2011:01) mengemukakan bahwa “suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.”

Pendekatan sistem yang lebih menekankan pada elemen atau komponennya mendefinisikan sistem sebagai berikut:

Menurut Puspita dan Anggadini (2011:02) mengemukakan bahwa “sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.”

Suatu sistem juga mempunyai beberapa karakteristik yaitu adanya tujuan sistem, batas sistem, subsistem, hubungan sistem, lingkungan sistem, *input*, proses dan *output*.

B. Pengertian Informasi

Menurut Darmawan dan Fauzi (2013:02) mengemukakan bahwa “informasi merupakan hasil dari pengolahan data, akan tetapi tidak semua hasil dari pengolahan tersebut bisa menjadi informasi, hasil yang tidak memberikan makna atau arti serta tidak bermanfaat bagi seseorang bukanlah merupakan informasi bagi orang tersebut”.

Agar informasi yang dihasilkan lebih berkualitas, maka informasi harus memenuhi ciri-ciri menurut Mc Leod dalam Darmawan dan Fauzi (2013:03), adalah:

1. Akurat

Informasi harus mencerminkan keadaan yang sebenarnya. Pengujian terhadap hal ini biasanya dilakukan melalui pengujian yang dilakukan oleh dua orang atau lebih yang berbeda dan apabila hasil pengujian tersebut menghasilkan hasil yang sama maka dianggap data tersebut akurat.

2. Tepat Waktu

Informasi harus tersedia atau ada pada saat informasi tersebut diperlukan, tidak besok atau tidak beberapa jam lagi.

3. Relevan

Informasi yang diberikan harus sesuai dengan yang dibutuhkan. Jika kebutuhan informasi ini untuk suatu organisasi maka informasi tersebut harus sesuai dengan kebutuhan informasi di berbagai tingkatan atau bagian yang ada dalam organisasi tersebut.

4. Lengkap

Informasi harus diberikan secara lengkap. Misalnya informasi tentang penjualan yang tidak ada bulannya atau tidak ada fakturnya.

C. Pengertian Sistem Informasi

Menurut Sutarman (2012:13) “ Sistem Informasi adalah system yang dapat didefinisikan dengan mengumpulkan, memproses, menyimpan, menganalisis,

menyebarkan informasi untuk tujuan tertentu. Seperti system lainnya, sebuah system informasi terdiri atas input (data,intruksi) dan output (laporan,kalkulasi).

Menurut Burch dan Grudnitski dalam Puspita dan Anggadini (2011:20) mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu:

1. Blok Masukan (*Input Block*)

Input yang mewakili data yang masuk ke dalam sistem informasi. Input disini dapat termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model (*Model Block*)

Terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan dalam basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*Technology Block*)

Teknologi merupakan “kotak alat” dalam sistem informasi yang digunakan untuk menerima input, menjadikan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran serta membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Basis Data (*Database Block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*Control Block*)

Beberapa pengendalian perlu dirancang untuk mencegah kerusakan, kegagalan sistem yang mungkin terjadi.

2.1.2. Konsep Dasar Web

A. Pengertian Internet

“*internet* merupakan kependekan dari kata “*internetwork*”, yang berarti rangkaian komputer yang terhubung menjadi beberapa rangkaian jaringan” (Irawan, 2011a:02).

Lahirnya internet dimulai pada sekitar tahun 1960-an sebagai suatu proyek dari *departement of defence* Amerika Serikat dan diberi nama ARPANET. Untuk dapat tersambung dengan internet, dibutuhkan sebuah modem yang menerjemahkan sebuah sinyal analog menjadi digital dan tampil pada layar komputer atau menghubungkan dengan ISP (*Internet Service Provider*) atau penyedia jasa layanan internet.

B. Pemrograman Web

Pemrograman *web* tidaklah sederhana karena banyak sekali teknologi yang ada di dalamnya. Teknologi ini terus berkembang dan bertambah banyak sehingga terdapat banyak istilah dan juga banyak bahasa-bahasa yang digunakan untuk membangun sebuah halaman *website*. Adapun yang termasuk dalam kategori

bahasa pemrograman *web* menurut Saputra (2012:02) diantaranya: HTML, PHP, ASP, XML, WML, PERL, CFM, JavaScript, CSS, JSP, Ruby dan Python.

C. Pengenalan HTML

HTML merupakan singkatan dari *Hypertext Markup Language*. Menurut Winarno dkk (2013:01) mengemukakan bahwa “HTML adalah bahasa *mark up* yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi di dalam sebuah *browser web* di internet”.

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, `tebal` 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, `bbb`).

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

D. 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 dimasukkan 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 open source*, 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:

1. PHP merupakan sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya. Tidak seperti halnya bahasa pemrograman aplikasi yang lainnya.
2. PHP dapat berjalan pada *web server* yang dirilis oleh Microsoft, seperti IIS atau PWS juga pada apache yang bersifat *open source*.
3. 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.
4. Jika dilihat dari segi pemahaman, PHP memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.
5. PHP dapat berjalan pada 3 *operating sistem*, yaitu: Linux, unix, dan windows, dan juga dapat dijalankan secara *runtime* pada suatu *console*.
6. 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 *web browser*. 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.

E. CSS (*Cascading Style Sheet*)

CSS atau memiliki kepanjangan *Cascading Style Sheet* merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur dan seragam (Saputra, 2012:27).

CSS merupakan program wajib yang harus dikuasai oleh setiap pembuat program (*web programmer*), terlebih lagi bagi para pendesain *web* (*Web designer*). CSS mampu diberbagai platform, maksudnya dapat dijalankan diberbagai macam sistem operasi dan *web browser*. Secara umum, yang dilakukan

oleh CSS adalah pengaturan *layout*, kerangka teks, gambar warna tebal, spasi dan lain sebagainya. CSS digunakan untuk mengatur tampilan, mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam.

F. JQuery

Menurut Irawan (2012b:03)“*JQuery* merupakan pustaka *JavaScript* yang dibangun untuk mempercepat dan memperingkas serta menyederhanakan manipulasi dokumen HTML, penanganan *event*, animasi dan interaksi ajaks untuk mempercepat pengembangan *web*”. Dengan *JQuery*, *develover* akan dimanjakan dengan satu pemrograman *JavaScript* yang sangat sederhana jika dibandingkan dengan *native JavaScript*.

G. Javascript

Menurut Irawan (2012b:02) : Javascript merupakan bahasa pemrograman web yang berbasis script. Menggunakan javascript dalam pemrograman web dapat menciptakan sebuah web dengan tampilan dan kinerja yang dinamis dan interaktif.

H. Xampp

Untuk membuat sebuah aplikasi berbasis web dengan menggunakan bahasa PHP, tentu saja diperlukan sebuah server web dan interpreter PHP. Server tidak harus sebuah computer khusus dengan kinerja tinggi dan berukuran sangat besa, tetapi bias dibuat dari PC yang mempunyai fungsi selayaknya sebuah web server, yaitu dengan menginstal paket XAMPP.

Menurut Riyanto (2011:04) “Xampp merupakan paket PHP dan MySQL berbasis open souce yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP”.

I. Basis Data

Basis data adalah Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah. (Fathansyah, 2012:02).

J. MySQL

Menurut Raharjo (2011:21) MySQL merupakan *software* RDBMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jurnal sangan besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).

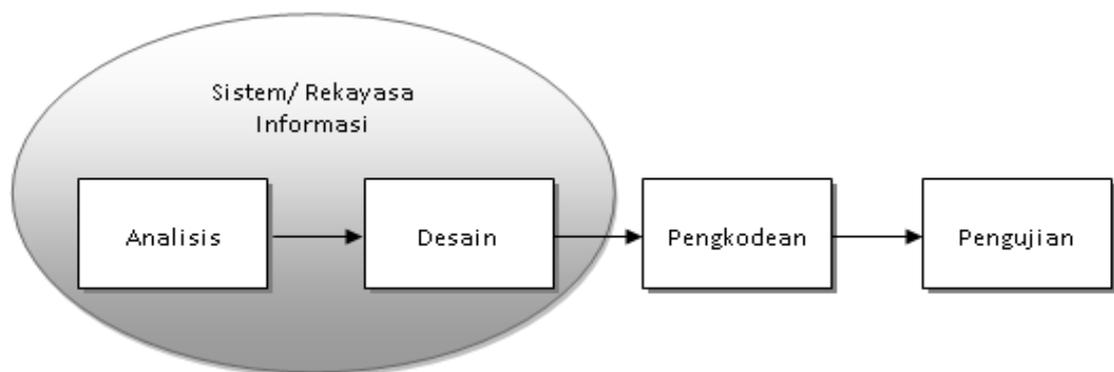
MySQL bekerja menggunakan bahasa *Stucture Query Language (SQL)* yang merupakan bahasa standar yang digunakan adalah untuk manipulasi *database*. Pada umumnya perintah yang sering digunakan adalah mengambil (*select*), menambah (*insert*), mengubah (*update*), dan menghapus (*delete*). Selain itu, *sql* juga menyediakan perintah untuk membuat *database*, *field*, ataupun *index* untuk menambah atau menghapus data.

2.1.3. Peralatan Pendukung Sistem

A. Model Waterfall

Menurut Rosa dan Shalahuddin (2015:31) “Model *waterfall* adalah model SDLC (*System Development Life Cycle*) yang paling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah”.

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*). Berikut adalah gambar model air terjun:



Gambar II.1 Ilustrasi model *waterfall*

Sumber: Rosa dan Shalahudin (2014:29)

Pembuatan model *waterfall* tentu tidak terlepas dari beberapa tahapan yang harus dikerjakan secara terstruktur. Lebih jelas tahapan-tahapan pembuatan model *waterfall* adalah sebagai berikut :

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara insentif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan Kode Program

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke pengguna. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

B. *Unified Modelling Language (UML)*

Pada Perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language (UML)*. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak.

Menurut Rosa dan Shalahuddin (2015:137) “UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”.

UML mempunyai sejumlah elemen grafis yang bisa dikombinasikan menjadi diagram. Karena ini merupakan sebuah bahasa, UML mempunyai aturan untuk menggabungkan elemen-elemen tersebut. Beberapa diagram UML yaitu:

1. *Activity Diagram*

Menurut Rosa dan Shalahudin (2015:161) “Activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

2. *Use Case Diagram*

Diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. (Rosa dan Shalahudin, 2015:155).

3. *Component Diagram*

Diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara komponen dalam sebuah sistem. Diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada di dalam sistem. (Rosa dan Shalahudin, 2015:148).

4. *Deployment Diagram*

Deployment Diagram menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. (Rosa dan Shalahudin, 2014:148).

C. Entity Relation Diagram (ERD)

Model ER menggambarkan data yang terlibat dalam organisasi, hubungan objek serta dapat digunakan untuk mengembangkan desain awal database. *Entity Relationship Diagram* menggambarkan lebih sistematis himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari ‘dunia nyata’. (Fathansyah, 2012:81)

Model ER sangat penting terutama perannya dalam desain basis data. Model ER menyediakan konsep yang memungkinkan untuk berpindah dari deskripsi apa yang pengguna inginkan pada basis data, untuk menjelaskan lebih rinci dan dapat diimplementasikan dalam DBMS (*Database Management System*).

Model ER digambarkan dengan ERD (*Entity Relationship Diagram*). ERD adalah kesatuan bentuk logika yang dipakai untuk analisa dan desain *database*. ERD menggambarkan arti dari aspek data. Di dalam pembuatan ERD perlu diperhatikan penentuan suatu konsep apakah merupakan suatu *entity*, *atribut*, atau *relationship*. Beberapa komponen ERD yang digunakan dalam merancang suatu sistem yaitu:

1. Entitas (*Entity*)

Entitas (*Entity*) adalah objek yang ada dan dapat dibedakan dengan objek lainnya. Sering lebih berguna untuk mengidentifikasi koleksi entitas-entitas yang serupa yang disebut himpunan entitas (*entity set*). Himpunan entitas adalah kumpulan entitas bertipe sama.

2. *Relationship*

Relationship adalah asosiasi di antara dua entitas atau lebih. Sebagaimana dengan entitas, kumpulan *relationship* serupa disebut himpunan *relationship* (*relationship set*). Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

3. Atribut

Atribut adalah karakteristik dari *entity* atau *relationship*, yang menyediakan penjelasan detail tentang *entity* atau *relationship* tersebut.

Atribut terdiri dari beberapa jenis yaitu:

a. *Key Attribute*

Atribut yang digunakan untuk menentukan suatu *entity* secara unik.

b. *Attribute Simple*

Atribut yang bernilai tunggal.

c. *Attribute Multivalued*

Atribut yang memiliki sekelompok nilai untuk setiap *instan entity*.

d. *Attribute Composite*

Suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu.

e. *Attribute Derivative*

Suatu atribut yang dihasilkan dari atribut yang lain

4. Kardinalitas (Derajat relasi)

Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari himpunan entitas yang satu ke himpunan entitas yang lain dan begitu juga sebaliknya. Kardinalitas di antara dua himpunan entitas (misalnya A dan B) dapat berupa:

a. *One to One* (1:1)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

b. *One to Many* (1:N)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

c. *Many to One* (N:1)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

d. *Many to Many* (N:N)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, demikian juga sebaliknya, di mana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

D. *Logical Relational Structures (LRS)*

Menurut Fathansyah, (2012:86) “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.

E. 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 menurut Simarmata (2010:321) diklasifikasikan menjadi :

1. Pengujian fungsional (*functional testing*)

Pada pengujian ini, perangkat lunak diuji untuk persyaratan fungsional.

Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan.

2. Pengujian tegangan (*stress testing*)

Pengujian ini berkaitan dengan kualitas aplikasi di dalam lingkungan.

Idenya adalah untuk menciptakan sebuah lingkungan yang lebih menuntut aplikasi, tidak seperti saat aplikasi dijalankan pada beban kerja normal.

3. Pengujian beban (*load testing*)

Pada pengujian beban, aplikasi akan diuji dengan beban berat atau masukan seperti yang terjadi pada pengujian situs web untuk mengetahui apakah aplikasi/situs gagal atau kinerjanya menurun.

4. Pengujian khusus (*ad-hoc testing*)

Jenis pengujian ini dilakukan tanpa penciptaan rencana pengujian (*test plan*) atau kasus pengujian (*test case*). Pengujian khusus membantu dalam menentukan lingkup dan durasi dari berbagai pengujian lainnya dan juga membantu para penguji dalam mempelajari aplikasi sebelum memulai pengujian lainnya.

5. Pengujian penyelidikan (*exploratory testing*)

Pengujian penyelidikan mirip dengan pengujian khusus dan dilakukan untuk mempelajari/mencari aplikasi. Pengujian penyelidikan perangkat lunak ini merupakan pendekatan yang menyenangkan untuk pengujian.

6. Pengujian usabilitas (*usability testing*)

Pengujian ini disebut juga sebagai pengujian untuk keakraban pengguna (*testing for user friendliness*). Pengujian ini dilakukan jika antarmuka pengguna dan aplikasinya penting dan harus spesifik untuk jenis pengguna tertentu.

7. Pengujian asap (*smoke testing*)

Jenis pengujian ini disebut juga pengujian kenormalan (*sanity testing*). Pengujian ini dilakukan untuk memeriksa apakah aplikasi tersebut sudah siap untuk pengujian yang lebih besar dan bekerja dengan baik tanpa cela sampai tingkat yang paling diharapkan.

8. Pengujian pemulihan (*recovery testing*)

Pengujian pemulihan (*recovery testing*) pada dasarnya dilakukan untuk memeriksa seberapa cepat dan baiknya aplikasi bisa pulih terhadap semua jenis *crash* atau kegagalan *hardware*, masalah bencana, dan lain-lain.

9. Pengujian volume (*volume testing*)

Pengujian volume dilakukan terhadap efisiensi dari aplikasi. Jumlah data yang besar diproses melalui aplikasi (yang sedang diuji) untuk memeriksa keterbatasan ekstrem dari system.

10. Pengujian domain (*domain testing*)

Pengujian domain merupakan penjelasan yang paling sering menjelaskan teknik pengujian.

11. Pengujian Skenario (*scenario testing*)

Pengujian scenario adalah pengujian yang realistik, kredibel dan memotivasi stakeholder, tantangan untuk program dan mempermudah penguji untuk melakukan evaluasi.

12. Pengujian regresi (*regression testing*)

Pengujian regresi adalah gaya pengujian yang berfokus pada pengujian ulang (*retesting*) setelah ada perubahan. Pada pengujian regresi berorientasi resiko (*risk-oriented regression testing*), daerah yang sama yang sudah diuji akan diuji lagi dalam pengujian yang berbeda(semakin kompleks).

13. Penerimaan pengguna (*user acceptance*)

Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan.

14. Pengujian alfa (*alpha testing*)

Pada jenis pengujian ini, pengguna akan diundang ke pusat pengembangan. Pengguna akan menggunakan aplikasi dan pengembang mencatat setiap masukan atau tindakan yang dilakukan oleh pengguna.

15. Pengujian beta (*beta testing*)

Pada jenis pengujian ini, perangkat lunak didistribusikan sebagai sebuah versi beta dengan pengguna yang menguji aplikasi di situs mereka.

2.2. Penelitian Terkait

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 SMK Industri Teknologi Nasional (ITENAS) Karawang.

Berikut beberapa tinjauan penelitian terdahulu (jurnal) yang dapat memperkuat alasan dibuatnya sistem informasi akademik pada SMK Industri Teknologi Nasional (ITENAS) Karawang, diantaranya:

Menurut Nisa, dkk dalam jurnalnya (2015:hal 142) teknologi komputerisasi belum diterapkan dengan baik, sehingga untuk penyampaian dan penyajian informasi akademik masih banyak mengalami masalah, pengolahan data siswa, data guru, dan data kehadiran siswa, pembagian kelas, penjadwalan hingga penilaian masih menggunakan buku atau arsip sehingga kurang efektif dan efisien karena belum adanya penyimpanan yg lebih terorganisir dengan baik sehingga memungkinkan terjadinya kerangkapan data. Untuk menangani latarbelakang permasalahan tersebut, penulis menerapkan

teknologi dengan system komputerisasi untuk membantu efesiensi kerja dengan menggunakan sistem komputerisasi berbasis *Web* atau *Internet* yang memudahkan penyampaian dan penyajian informasi dengan lengkap dan akurat.

Menurut Kurnia, dkk dalam jurnalnya (2012: hal 1) untuk mendapatkan informasi mengenai jadwal pelajaran, nilai hasil ujian dan tugas masih bersifat manual. Hal ini dapat memperlambat dan menghambat informasi-informasi bagi siswa dan guru, karena selama ini pengolahan akademik yang dilakukan dan penyajian informasi berada di sekolah. Untuk mengurangi kelemahan SMK Ciledug Al-Musaddadiyah garut perlu memiliki sebuah sistem yang dapat memudahkan guru dan siswa dalam memberikan dan mendapatkan informasi secara mudah dan cepat. Karena alasan itulah memutuskan tujuan dalam penelitian tugas akhir ini untuk merancang sistem informasi akademik nilai siswa secara online yang nantinya akan digunakan sebagai alat bantu dalam pengelolaan informasi pada SMK Ciledug

Berdasarkan tinjauan penelitian terkait diatas, dengan menggunakan dan memanfaatkan Sistem Informasi Akademik Berbasis *Web* diharapkan guru, siswa, orang tua siswa, dan calon siswa SMK Industri Teknologi Nasional memperoleh beberapa keuntungan diantaranya kemudahan mencari informasi, akses informasi menjadi cepat, perhitungan menjadi akurat, mengurangi penggunaan kertas. Banyak biaya yang dapat dipangkas, dan fleksibilitas bertambah. Dengan pemanfaatan informasi berbasis *web* di harapkan dapat meningkatkan kualitas pendidikan di Indonesia.