BAB II
LANDASAN TEORI

2.1 Konsep Dasar Informasi

2.1.1 Definisi Data

Sumber informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum atau item. Menurut McLeod dalam bukunya Yakub (Yakub, 2012:5) "Data adalah deskripsi kenyataan yang menggambarkan adanya suatu kejadian (*event*), data terdiri dari fakta (*fact*) dan angka yang secara relatif tidak berarti bagi pemakai". Menurut Tata Sutabri (2012:2) , data adalah bahan mentah yang diproses untuk menyajikan informasi.

Menurut The Liang Gie dalam Tata Sutabri (2012:2), bahwa data adalah hal, peristiwa, atau kenyataan lain apapun yang mengandung sesuatu pengetahuan untuk dijadikan dasar guna penyusunan keterangan, pembuatan kesimpulan, atau penetapan keputusan.

2.1.2 Definisi Informasi

Informasi ibarat darah yang mengalir di dalam tubuh manusia, maksud dari kalimat tersebut yaitu bahwa informasi sangat penting pada suatu organisasi. Informasi (information) dapat didefinisikan sebagai berikut:

a. Menurut Mustakini (2009:36), "Informasi adalah data yang diolah menjadi bentuk yang berguna bagi pemakainya".
b. Menurut McLeod dalam Yakub (2012:8), "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna bagi penerimanya".
c. Berdasarkan pendapat para ahli yang dikemukakan di atas dapat ditarik kesimpulan bahwa informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang

bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi.

2.1.2.1 Kualitas Informasi

Kualitas informasi sangat dipengaruhi atau ditentukan oleh tiga hal pokok, yaitu akurasi (*accuracy*), relevansi (*relevancy*), dan tepat waktu (*timeliness*). (Agus Mulyanto, 2009 :247).

2.1.2.2 Nilai Informasi

Parameter untuk mengukur nilai sebuah informasi (*value of information*) ditentukan dari dua hal pokok yaitu manfaat (*benefit*) dan biaya (*cost*). Namun, dalam kenyataannya informasi yang biaya untuk mendapatkannya tinggi belum tentu memiliki manfaat yang tinggi pula. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkannya dan sebagian besar informasi tidak dapat tepat ditaksir keuntungannya dengan satuan nilai uang, tetapi dapat ditaksir nilai efektivitasnya.(Agus Mulyanto, 2009 : 247).

2.2 Konsep Dasar Sistem Informasi

2.2.1 Definisi Sistem Informasi

Terdapat berbagai macam pengertian sistem informasi menurut beberapa ahli, diantaranya sebagai berikut :

Menurut Mulyanto (2009:29), “Sistem informasi adalah suatu komponen yang terdiri dari manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai suatu tujuan”.

Menurut fitria dan afrida (2015:33), “Sistem informasi merupakan suatu sistem yang tujuannya menghasilkan informasi.”

Dari pendapat yang dikemukakan di atas dapat disimpulkan bahwa informasi adalah gabungan dari orang, *hardware*, *software*, jaringan komunikasi, sumber daya data, dan kebijakan dan prosedur yang menyimpan, mengumpulkan (mendapatkan kembali), memproses, dan mendistribusikan informasi untuk mendukung pengambilan dan pengontrolan keputusan dalam suatu organisasi.

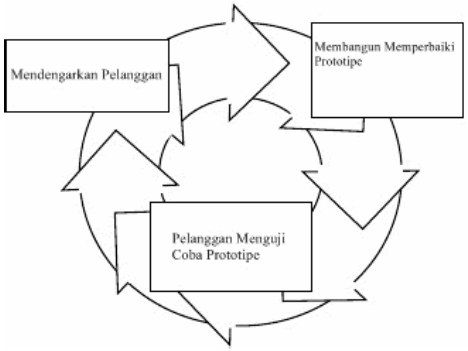
2.2.2 Komponen Sistem Informasi

John Burch dan Gary Grudnitski dalam Jogiyanto , mengemukakan bahwa "Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (building block). Sebagai suatu sistem, blok bangunan tersebut masing-masing berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk

2.3 Teori Dasar Pengembangan Sistem

Sutanta (2004), sistem dapat didefinisikan sebagai sekumpulan hal atau kegiatan atau elemen atau subsistem yang saling bekerja sama atau yang dihubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan.

Prototype dapat hanya mencakup fungsi-fungsi yang paling penting atau mencakup seluruh sistem. Terlebih dahulu peneliti melakukan analisa terhadap prosedur yang berjalan, analisis kebutuhan sistem kemudian menyiapkan rancangan perangkat lunak yang akan digunakan dalam prosedur yang baru. Model *prototype* dapat dilihat pada gambar 3.1.



Gambar 2.1 Model Prototype

Dalam metode *prototype*/*prototyping*, perangkat lunak yang dihasilkan kemudian dipresentikan kepada klien, dan klien tersebut diberikan kesempatan untuk memberikan masukan dan kritikan, sehingga perangkat lunak / *software* yang dihasilkan sesuai dengan kebutuhan dan keinginan pelanggan. Perubahan perangkat lunak dapat dilakukan berkali-kali hingga dicapai kesepakatan bentuk dari *software* yang akan dikembangkan

2.4 Sejarah UML

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian diikuti dengan perkembangan bahasa pemrograman berorientasi objek yang lainnya seperti C objek, C++, Eiffel, dan CLOS.

Sekitar lima tahun setelah Smalltalk berkembang, maka berkembang pula metode pengembangan berorientasi objek. Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep, seperti konsep *Object ModelingTechnique (OMT)* dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities, Collaborators (CRC)* dari Rebecca Wirfs-Brock (1990), konseppemikiran Ivar Jacobson, dan beberapa konsep lainnya dimana James R. Rumbaigh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama Rational Software Corporation menghasilkan bahasa yang disebut dengan *Unified Modeling Language (UML).*

Pada tahun 1996, *Object Management Group (OMG)* mengajukan proposal agar adanya standarisasi pemodelan berorientasi objek dan pada bulan September 1997 *UML* diakomodasi oleh *OMG* sehingga sampai saat ini *UML* telah memberikan kontribusinya yang cukup besar di dalam

metodologi berorientasi objek dan hal-hal yang terkait di dalamnya. (Rosa A.S dan M. Shalahudin, 2014:138).

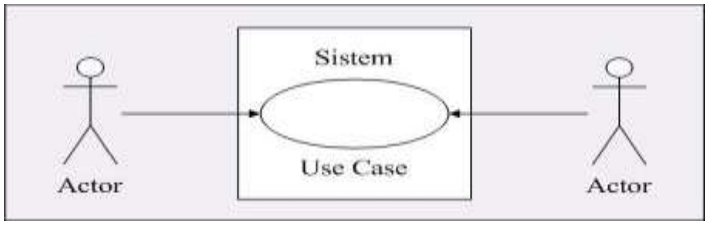
2.4.1 Diagram UML

Rosa A.S dan M. Shalahudin (2014:140), pada *UML* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

1. *Structure diagram*, yaitu kumpulan diagram yang digunakan untukmenggambarkan suatu struktur statis dari sistem yang dimodelkan. *Structurediagram* terdiri dari *class diagram, object diagram, component diagram, composite structure diagram, package diagram* dan *deployment diagram.*
2. *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. *Behavior* diagram terdiri dari *Use case diagram, Activity diagram, State Machine System.*
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. *Interaction diagram* terdiri dari *Sequence Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.*

2.5 *Use Case* Diagram

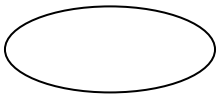
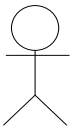
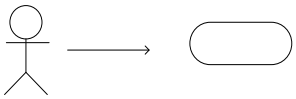
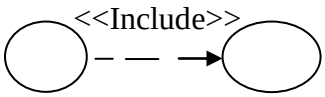
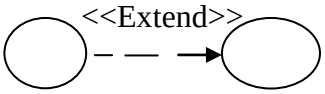
Menurut Nugroho (2010) menyatakan bahwa *Use case* adalah mendeskripsikan fungsi sebuah sistem dari sisi pengguna. *Use case* juga digunakan untuk membentuk perilaku (*behaviour*) sistem yang akan dibuat. Sebuah *use case* menggambarkan sebuah interaksi antara pengguna (aktor) dengan sistem yang ada.



Gambar 2.2 *Use case* (Nugroho, 2010)

Simbol-simbol *use case diagram* dapat dilihat pada tabel II-1 dibawah ini

Tabel 2.1 Simbol Diagram *Use case*

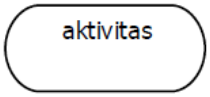
No	Simbol	Keterangan
1		*Usecase* : menggambarkan bagaimana seseorang akan menggunakan atau memanfaatkan sistem.
2		Aktor : seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan.
3		Relasi: sebagai penghubung antara aktor-*usecase*, *usecase-usecase* dan lain-lain.
4		Relasi Asosiasi: relasi terjadi antara aktor dengan *usecase* biasanya berupa garis lurus dengan kepala panah disalah satu ujungnya.
5		*Include relationship* (relasi cakupan) : memungkinkan suatu *usecase* untuk menggunakan fungsionalitas yang disediakan oleh *usecase* yang lainnya.
6		*Extend relationship* : memungkinkan *usecase* memiliki kemungkinan untuk memperluas fungsionalitas yang disediakan oleh *usecase* yang lainnya.

2.6 *Activity* Diagram

Rosa dan M. Shalahudin (2014:161), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuahsebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas :

Tabel 2.2 *Activity Diagram*

SIMBOL	KETERANGAN
Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, sebuah diagram aktivitas biasanya diawali dengan kata kerja.
Penggabungan / ***join***	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Percabangan/*decision*	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

Swimlane atau	Memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.
Join 	digunakan untuk menunjukkan kegiatan yang digabungkan

Sumber: Rosa A.S dan M. Shalahuddin (2014 : 162)

2.7 *World Wide Web*

2.7.1 *Internet*

Internet merupakan contoh jaringan terbesar yang menghubungkan jutaan komputer yang tersebar diseluruh penjuru dunia dan tak terikat pada satu organisasi. *Protocol TCP/IP (Transmission Control Protocol/Internet Protokol)* untuk mengoneksi jutaan Komputer di *internet*. Protokol ini mensyaratkan bahwa setiap Komputer didalam jaringan internet harus memiliki identitas yang unik yang dinamakan nomor atau alamat *IP* yang dinyatakan dengan nama domain. HTTP *(Hypertext Transfer Protocol)*, merupakan protokol yang digunakan *web* yang berjalan pada TCP/IP (Nugroho, 2008).

2.7.2 Pengertian *Web*

Web adalah suatu sistem informasi yang berbasis grafis yang menampilkan data berupa teks, gambar, animasi, video dan bunyi, yang di antara data tersebut saling berhubungan dengan yang lainya di *internet* (Nugroho, 2008:47)**.**

2.8 Pengertian Teknik Normalisasi

Normalisasi adalah suatu teknik yang menstrukturkan memecah/mendekomposisi data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. (Sutanta, 2004).

Proses normalisasi, ada beberapa istilah yang akan dipakai yaitu:

a) *Entity*

   Entity adalah konsep informasi yang direkam, meliputi orang, kejadian dan tempat.

b) *Atribut* atau *Field*

   Atribut atau *Field* adalah sesuatu yang mewakili *entity*.

c) Data *Value*

   Data *Value* adalah informasi yang tersimpan dalam setiap *atribut*.

d) *Record*

   Record adalah kumpulan atribut yang saling berkaitan satu dengan yang lain dan menginformasikan suatu *entity* secara lengkap.

e) *File*

   File adalah kumpulan *record* yang mempunyai panjang atribut yang sama tetapi berbeda data *value*nya.

f) Basis data atau *Database*

   Database adalah kumpulan *file* satu dengan *file* yang lainnya yang membentuk satu informasi sistem secara keseluruhan.

Proses normalisasi, *field* kunci memegang peranan yang sangat penting dalam pembuatan tabel yang berisi *entity* dan relasinya. *Field* kunci merupakan satu *field* atau satu set *field* yang terdapat dalam satu *file* yang merupakan kunci dan mewakili *record*. Kunci disini akan sangat penting apabila dalam program nanti terdapat fasilitas pencarian, karena *field* yang merupakan kunci akan menjadi penentu dalam pencarian program. *Field* kunci dibagi menjadi beberapa bagian yaitu:

a) Kunci Kandidat

Kunci Kandidat adalah satu *atribut* atau *field* yang mengidentifikasikan secara unik dari suatu kejadian yang sifatnya khusus dari suatu *entity*.

b) Kunci *Primer*

Kunci *Primer* adalah kunci kandidat yang dipilih untuk mewakili setiap kejadian dari suatu *entity*.

c) Kunci Alternatif

Kunci Alternatif adalah kunci kandidat yang tidak dipakai sebagai kunci *primer*.

d) Kunci Tamu

Kunci Tamu adalah kunci primer yang ditempatkan pada file lain dan biasanya menunjukan dan melengkapi suatu hubungan antara file satu dengan file yang lainnya.

Proses normalisasi juga membutuhkan beberapa tahap sebelum nantinya akan diimplementasikan dalam program. Tahap-tahap normalisasi adalah :

a) Bentuk Tidak Normal

Bentuk tidak normal adalah suatu bentuk dimana semua data dikumpulkan apa adanya tanpa mengikuti atura-aturan tertentu. Bisa jadi data yang dikumpulkan akan tidak lengkap dan terjadi duplikasi data.

b) Bentuk Normal Pertama

Bentuk Normalisasi Pertama adalah suatu bentuk dimana data yang dikumpulkan menjadi satu *field* yang sifatnya tidak akan berulang dan tiap *field* hanya mempunyai satu pengertian.

c) Bentuk Normal Kedua

Bentuk Normal Kedua adalah suatu bentuk yang memenuhi syarat-syarat, yaitu:

1) Sudah memenuhi kriteria sebagai bentuk normal pertama
2) *Field* yang bukan kunci tergantung secara fungsi pada kunci primer.

d) Bentuk Normal Ketiga

Bentuk Normal Ketiga adalah suatu bentuk yang memenuhi syarat-syarat, yaitu:

1. Relasi antar *file* sudah merupakan bentuk normal kedua
2. *Field* yang bukan kunci tergantung secara fungsi pada kunci primer.

e) Bentuk Normal Keempat

Bentuk Normal Keempat adalah suatu bentuk pengujian dalam arti bahwa kepastian isi *file* atau tabel dan hubungan tabel satu dengan tabel yang lain.

2.9 PHP

PHP (*PHP Hypertext Preprocessor)* yang digunakan sebagai bahasa *script serve-side* dalam pengembangan Web yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan Web dapat dibuat dinamis sehingga *maintenance* situs *web* tersebut lebih mudah dan efisien. PHP merupakan *software Open-Source* yang disebarkan dan dilisensikan secara gratis serta dapat di-*download* secara bebas dari situs resminya yaitu : http://www.php.net. PHP yang ditulis dengan menggunakan bahasa C dapat dikembangkan sendiri dengan menambah fungsi-fungsi baru. PHP adalah bahasa yang sangat popular. Sebagian besar sintaks PHP mirip dengan sintaks dalam *bahasa C* dan *Perl*.

Kelebihan PHP adalah tidak dimiliki oleh bahasa *script* sejenis. PHP difokuskan pada pembuatan *script server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh GCI, seperti mengumpulkan data dari form, menghasilkan isi halaman *web* dinamis, dan kemampuan mengirim serta menerima *cookies* bahkan lebih daripada kemampuan CGI. PHP dapat digunakan pada semua sistem operasi, antara lain Linux, Unix, Microsoft Windows, Macintosh. PHP juga mendukung banyak *Web Server*, seperti Apache, *Microsoft Internet Information server* (MIIS), *Personal Web Server* (PWS), dan masih banyak lagi lainnya, bahkan PHP dapat bekerja sebagai suatu CGI *processor*. PHP tidak terbatas pada hasil keluaran HTML (*HyperText Markup Languages*).

PHP juga memiliki kemampuan untuk mengolah keluaran gambar, *file* PDF, dan *movies Flash*. PHP juga dapat menghasilkan teks seperti XHTML dan file XML lainnya.

2.9.1 Sintaks PHP

Sintaks program / *script PHP* ditulis dalam apitan tanda khusus PHP. Ada empat macam pasangan tag PHP yang dapat digunakan untuk menandai blok script PHP.

<?php …?>

<script language = “PHP”> … </script>

<? … ?>

<% … %>

Cara 1 dan 2 merupakan cara yang paling umum digunakan sekalipun. cara 3 tampak lebih praktis karena cara 3 tidak selalu diaktifkan pada konfigurasi *file* php. Ini yang terdapat pada direktori: C:\Program Files\Apache Software Foundation\Apache2.2\PHP. Cara 4 juga dimungkinkan sebagai kemudahan bagi Anda yang sudah terbiasa dengan ASP (*Active Server Pages*). Namun, bila itu tidak dikenal, maka harus dilakukan pengaktifan pada *file* konfigurasi php.ini. Pengaktifan yang dilakukan pada *file* php.ini terdapat pada baris berikut: Language options Allow ASP –style <% %> tags. Asp_tags = off Ubah *Off* menjadi On, kemudian simpan dan restart kembali web *server* Anda.

2.9.2 MySQL (*My Structure Query language*)

MySQL adalah sebuah program pembuat database yang bersifat *open source* artinya siapa saja boleh menggunakannya dan dapat dijalankan pada semua *platform* baik *windows* maupun *linux*. Selain itu, MySQL juga merupakan prigram pengakses database yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *multiuser* (banyak pengguna).

Kelebihan lain dari MySQL adalah menggunakan bahasa query standar yang memiliki SQL (*Structure Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengaksesan database seperti *Oracle*, *Progres SQL*, *SQL Server*, dan lain-lain.

Sebagai program pembangunan *database*, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain (*interface*). MySQL dapat didukung oleh hamper semua program aplikasi baik yang open source seperti PHP maupun tidak, yang ada *platform windows* seperti *Visual Basic*, *Delphi* dan lain sebagainya.

2.9.3 Macromedia Dreamweaver 8

Saat ini *Dremweaver* merupakan *software* utama yang digunakan oleh *web server desainer* maupun *web programmer* dalam membangun suatu situs *web*. *Dreamweaver* memberikan kemudahan bagi penggunanya dalam menentukan ruang kerja yang diinginkan. Tipe ruang kerja, fasilitas dan kemampuan *dreamweaver* mampu meningkatkan produktivitas dan efektivitas dalam desain maupun dalam membangun sebuah situs *web*.

Versi terbaru *Macromedia Dreamweaver* di tahun 2013 adalah *Macromedia Dreamweaver* 8 yang merupakan penyempurnaan dari versi sebelumnya dan tentu saja semakin mudah dalam penggunaannya. Fasilitas terbaru dari *Macromedia Dreamweaver* 8 adalah *Zoom Tool and Guides*, *Panel CSS* yang baru, *Code Collapse*, *Coding Toolbar*, *Insert Flash Video* dan tampilan