

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

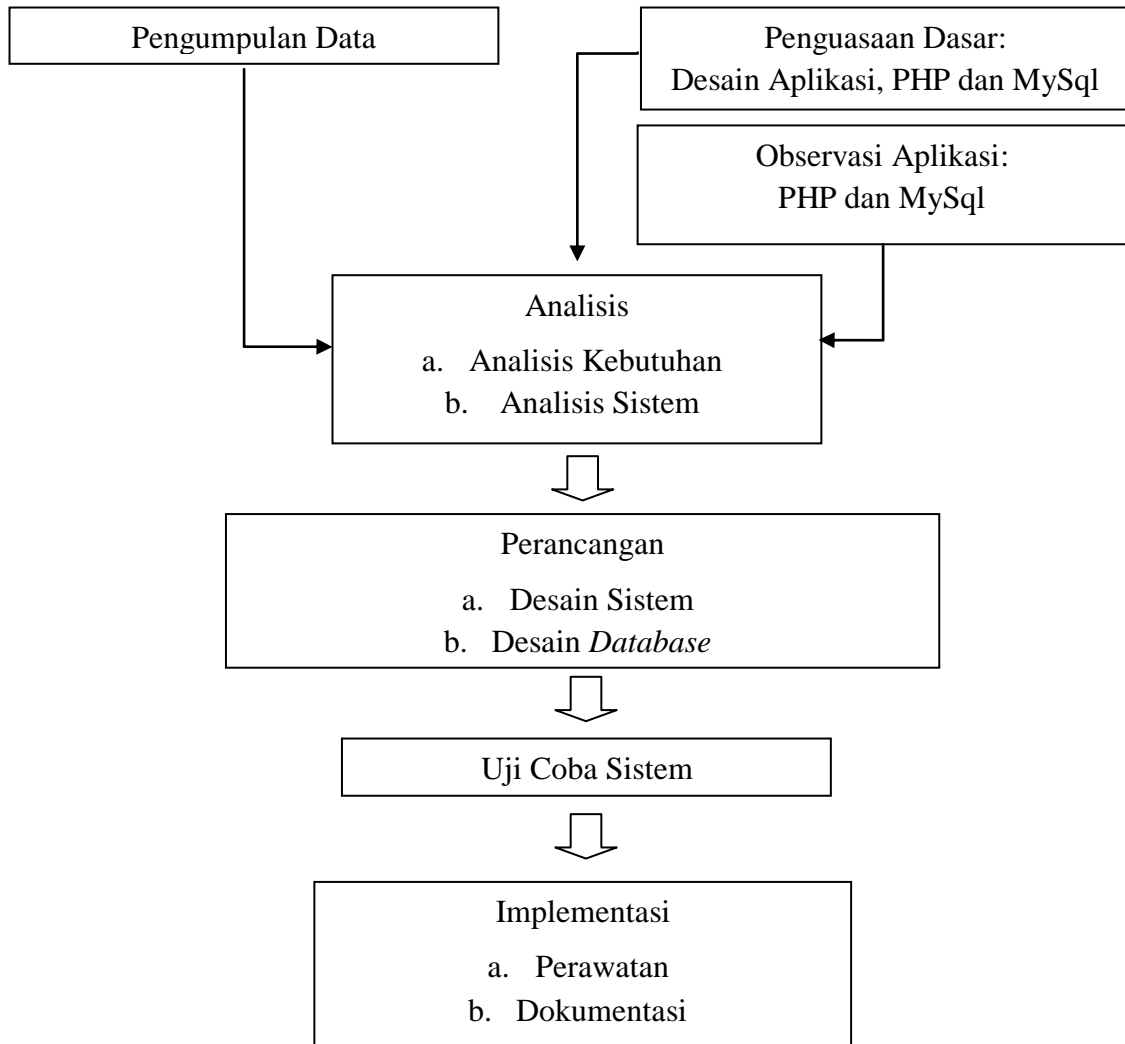
Sistem informasi yang digunakan di pusat penjaminan mutu Universitas Sahid Surakarta ini berfungsi sebagai media penyampaian informasi terkait penjaminan mutu di Universitas Sahid Surakarta, dimana sistem yang berjalan saat ini masih dilakukan secara manual. Tujuan dari penelitian ini adalah mengetahui sejauh mana sistem yang sudah berjalan saat ini, lalu mengembangkan sistem informasi penjaminan mutu berbasis *web* di Pusat Penjaminan Mutu Universitas Sahid Surakarta. maka pada perancangan sistem yang baru semua akan dilakukan dengan cara terkomputerisasi dalam satu sistem informasi yang berbasis *web*. Sistem informasi yang terkomputerisasi dimaksudkan agar memberi kemudahan kepada *user* dan pusat penjaminan mutu. Sehingga penyampaian informasi tentang penjaminan mutu dapat dilakukan dengan mudah.

Mengutip pendapat Yanuar Nur F.P (2012) keberadaan sistem informasi data merupakan hal yang penting mengingat masih ada pencatatan kegiatan secara manual dan sederhana, sistem pencatatan tersebut berdampak pada lamanya waktu yang digunakan dalam melakukan proses pendataan dan pencarian.

Heri Kusharyanto (2013) menyatakan bahwa pengolahan data secara manual memiliki beberapa kelemahan yaitu sering terjadi kesalahan dan membutuhkan waktu yang lama dalam pencatatan serta pencarian data, maka perlu dibangun program aplikasi agar dapat mempermudah pengolahan data dan pencarian informasi.

Mengutip pendapat Josua Filemon Johanes (2010) bahwa pengembangan sistem informasi bertujuan agar diperoleh sebuah inovasi baru, sehingga efek dari pengembangan tersebut adalah ketertarikan *user* untuk menggunakan aplikasi.

2.2 Kerangka Pemikiran



Gambar 2.1 Kerangka Pemikiran

Keterangan Bagan Alir Kerangka Pemikiran

1. Pengumpulan data
Mengumpulkan semua data melalui *interview*, observasi dan studi literatur.
2. Penguasaan Dasar
Melakukan beberapa percobaan membuat aplikasi sederhana dengan tujuan agar dapat lebih menguasai bahasan pemrograman PHP.

3. Observasi Aplikasi

Mengamati, membandingkan dan mempelajari aplikasi yang berjalan saat ini, dan aplikasi lain baik dari karya ilmiah, buku, *internet* yang dapat dijadikan referensi.

4. Analisis

1) Analisis Kebutuhan

Menganalisa dan merancang sistem yang akan dibangun seperti apa, bagaimana desainnya, dan apa saja isinya sesuai dengan kebutuhan *user*.

2) Analisis Sistem

Menganalisa apa saja hal-hal yang akan dibutuhkan dalam pembuatan sistem informasi pusat penjaminan mutu Universitas Sahid Surakarta sehingga sistem ini nantinya dapat memberikan solusi atas perumusan masalah.

5. Perancangan

1) Perancangan *Database*

Membuat *database* dari data-data yang didapatkan sesuai dengan kebutuhan.

2) Perancangan Sistem

Perancangan sistem informasi pusat penjaminan mutu Universitas Sahid Surakarta, Membuat aplikasi dengan bahasa PHP berdasarkan *database* yang telah dibuat

6. Uji Coba Sistem

Pengujian dilakukan pada pusat penjaminan mutu Universitas Sahid Surakarta dengan penguji dari pengguna sistem (*Admin* dan *staf*, dan *pengunjung*) dengan tujuan untuk mengetahui sistem tersebut sudah menjawab perumusan masalah atau belum.

7. Perawatan Sistem dan Dokumentasi

Tahap akhir dimana sistem telah di uji coba dan siap digunakan pusat penjaminan mutu Universitas Sahid Surakarta, setelah itu adalah perawatan yang dilakukan berkala oleh *admin* sisten informasi pusat penjaminan mutu

Universitas Sahid Surakarta dengan panduan dari dokumentasi keseluruhan kegiatan pengembangan sistem informasi.

2.3 Pengertian Sistem

Hanif Al Fatta (2007:3) mengutip pendapat Murdick dan Ross (1993) bahwa definisi sistem adalah seperangkat elemen yang digabungkan satu dengan yang lainnya untuk suatu tujuan bersama.

Jogiyanto (2005:2) menyatakan sistem adalah sebagai suatu kumpulan komponen yang berinteraksi membentuk suatu kesatuan dan keutuhan yang kompleks di dalam tingkat tertentu untuk mengejar tujuan yang umum.

Menurut Zulkifli Amsyah (2000:4) Sistem adalah elemen-elemen yang saling berhubungan membentuk suatu kesatuan atau organisasi. Adapun pengertian sistem menurut Jogiyanto (2005:1) adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu. Menurut L. Ackof, sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya (Abdul Kadir, 2008:54).

Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung satu sama lain serta saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

2.3.1 Pengertian Informasi

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih bermanfaat bagi yang menerima (Andi Kristanto, 2003;6). Jadi ada suatu proses transformasi dari data menjadi suatu informasi, yaitu *input proses output*. Informasi yang baik adalah informasi yang berkualitas, yaitu informasi harus akurat (bebas dari kesalahan dan tidak menyesatkan), tepat waktu (*up to date*), dan relevan (mempunyai manfaat bagi penerima informasi).

Informasi adalah sekumpulan fakta (data) yang diorganisasikan dengan cara tertentu sehingga mereka mempunyai arti bagi si penerima. Sebagai contoh, apabila kita memasukkan jumlah gaji dengan jumlah jam bekerja, kita akan mendapatkan informasi yang berguna. Dengan kata lain, informasi datang dari data yang akan diproses (Sutarman, 2009 : 14).

2.3.2 Pengertian Sistem Informasi

Sistem informasi atau disebut juga dengan sistem proses (*processing systems*) atau (*information-generating system*). Sistem Informasi didefinisikan oleh Leitch dan Davis dalam (Jogiyanto, 2005:11) Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Menurut Gede Karya (2004:01) definisi dari sistem informasi adalah sekumpulan komponen yang saling berhubungan yang mengumpulkan, memproses, menyimpan dan mendistribusikan informasi mendukung pembuatan keputusan dan pengendalian suatu organisasi.

Sistem Informasi ini mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan informasi untuk tujuan tertentu. Seperti sistem lainnya, sebuah sistem informasi terdiri atas *input* (data, instruksi) dan *output* (laporan, kalkulasi). Sistem Informasi memproses *input* dan menghasilkan *output* yang dikirim kepada pengguna atau sistem yang lainnya. Mekanisme timbal balik yang mengontrol operasi pun bisa dimasukkan. Seperti sistem lainnya, sebuah sistem informasi beroperasi di dalam sebuah lingkungan. Dalam mempelajari sistem informasi, perlu diketahui mengenai perbedaan data, informasi, dan pengetahuan. (Sutarman, 2009 : 13).

Pada beberapa pendapat yang ada maka diambil kesimpulan bahwa pengertian sistem informasi adalah suatu kumpulan fungsi-fungsi yang secara baku dan terstruktur bekerja secara bersama-sama dan simultan, atau dengan kata lain bahwa sistem informasi adalah suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai

tujuan yaitu menyajikan informasi. Dengan demikian, sistem informasi berdasarkan konsep IPO.

2.4 Pengertian PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis (M. Rusdyanto Arief, 2011:43). Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah selanjutnya akan dieksekusi di *server* kemudian hasilnya akan dikirimkan ke *Browser* dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh *user* sehingga keamanan halaman *web* lebih terjamin. PHP dirancang untuk membentuk halaman *web* yang dinamis, yaitu halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*.

PHP merupakan bahasa standar yang digunakan dalam dunia *website*. PHP adalah bahasa pemrograman yang berbentuk skrip yang diletakan di dalam *server web*. Jika lihat sejarah mulanya PHP diciptakan dari ide Rasmus Lerdof untuk kebutuhan pribadinya. Skrip tersebut sebenarnya dimaksudkan untuk digunakan sebagai keperluan membuat *website* pribadi. Akan tetapi kemudian dikembangkan lagi sehingga menjadi bahasa yang disebut "*Personal Home Page*". Inilah awal mula munculnya PHP sampai saat ini. PHP dirancang untuk membentuk *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, menampilkan isi *database* ke halaman *web*. Pada prinsipnya, PHP mempunyai fungsi yang sama dengan *script* seperti ASP Cold Fusion, ataupun Perl (Bunafit Nugroho, 2008 : 113).

2.5 Pengertian CSS

CSS adalah sebuah fitur yang diperkenalkan sejak HTML versi 4.0 dan berfungsi untuk menangani masalah tampilan pada HTML seperti jenis, ukuran dan warna *font*, posisi teks, batas tulisan atau *margin*, warna *background*, dan sebagainya.

Dari sisi manajemen dan perawatan, penggunaan CSS dipandang lebih praktis, karena para *web developer* tidak perlu membuka setiap *file* dalam sebuah

situs untuk melakukan perubahan. Hal penting yang perlu diperhatikan adalah cara meletakkan CSS dan juga bahasa berbasis *web* lain untuk memudahkan manajemen *file*, *editing*, dan *maintenance*. Banyak diantara para *programmer web* yang belum menyadari aspek – aspek penting ini menyisipkan CSS, *JavaScript*, *VBScript*, PHP, maupun ASP langsung kedalam dokumen HTML (*embedded script*) (Madcoms, 2008 : 100).

2.6 Pengertian Basis Data

Basis data adalah kumpulan dari item data yang saling berhubungan satu dengan lainnya, yang diorganisasikan berdasar sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan menggunakan *software* tertentu digunakan untuk melakukan manipulasi data untuk tujuan tertentu.

Suatu basis data adalah koleksi data yang bisa mencari secara menyeluruh dan sistematis memelihara informasi (Janner Simarmata, 2011).

Menurut Fathansyah (2007:2) Basis data terdiri dari 2 kata yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat. Sedangkan Data adalah representasi fakta yang mewakili suatu objek.

Basis data sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

- a. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
- b. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
- c. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Menurut Diar Puji Oktavian (2010 : 62) definisi *database* adalah sekumpulan data dan prosedur yang memiliki struktur sedemikian rupa sehingga mudah dalam menyimpan, mengatur dan menampilkan data. Banyak program *database* yang tersedia, diantaranya adalah *Oracle*, *MySQL*, *MSSQL*, *PostgreSQL*,

Paradox, Foxpro dan lain – lain. *Database* terbentuk dari beberapa komponen, yaitu :

1) *Table*

Table atau tabel adalah sekumpulan data dengan struktur yang sedemikian rupa, terbentuk dari *record* dan *field*.

2) *Record*

Record adalah sekumpulan *field* yang membentuk suatu objek tertentu.

3) *Field*

Field adalah atribut dari objek yang memiliki tipe data tertentu.

2.7 Pengertian Website

Website merupakan kumpulan dari halaman *web* yang sudah dipublikasikan di jaringan *internet* dan memiliki domain/URL yang dapat diakses semua pengguna *internet* dengan cara mengetikkan alamatnya (M. Rusdyanto Arief, 2011:8). Menurut Simarmata (2010) *website* adalah sebuah sistem informasi yang disajikan dalam bentuk teks, gambar, suara dan lain-lain yang tersimpan dalam sebuah *server web internet* yang disajikan dalam bentuk *hiperteks web browser*.

Website merupakan program yang berfungsi untuk menampilkan dokumen-dokumen *web* dalam format HTML (M. Rusdyanto Arief, 2011:19). Halaman *web* yang dibuat ditampilkan sangat tergantung pada *web engine* yang digunakan oleh masing-masing *browser*. Semua jenis *web browser* yang ada saat ini mengikuti standarisasi yang dibuat oleh W3C yang merupakan badan independen yang mengurus semua hal yang berkaitan dengan *web* di dunia. Sebagai contoh *web browser* yang cukup populer beserta *web engine* yang digunakan :

- 1) *Web Engine Webkit : Safari, Google Chrome.*
- 2) *Web Engine Trident : Microsoft Internet Explorer, Maxthon.*
- 3) *Web Engine Gecko : Mozilla Firefox.*
- 4) *Web Engine Presto : Opera.*

2.8 Pengertian Web Server

Web server adalah sebuah perangkat lunak *server* yang berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML. *Web server* juga merupakan sebuah komputer yang menyediakan layanan untuk *internet*, *server* disebut juga dengan *host*. Agar anda dapat memasukkan *web* yang anda rancang ke dalam *internet*, maka anda harus memiliki ruangan terlebih dahulu dalam *internet*, dan ruangan ini disediakan oleh *server*. Itulah yang disebut *web server*.

2.9 Pengertian Adobe Dreamweaver

Adobe dreamweaver merupakan salah satu program aplikasi yang digunakan untuk membangun sebuah *website*, baik secara grafik maupun dengan menuliskan kode sumber secara langsung (Maria Agustina S, 2010:2).

Adobe dreamweaver memudahkan pengembang *website* untuk mengelola halaman-halaman *website* dan aset-asetnya, baik gambar , animasi flash, video, suara dan lain sebagainya. Selain itu adobe dreamweaver menyediakan fasilitas untuk melakukan pemrograman scripting, baik ASP, JSP, PHP, *JavaScript (js)*, *Cold Fusion*, *CSS*, *XML* dan lainnya.

2.10 Pengertian MySQL

SQL merupakan kepanjangan dari *Structured Query Language*, yang merupakan bahasa terstruktur yang khusus digunakan untuk mengolah *database*. SQL pertama kali didefinisikan oleh *ANSI* pada tahun 1986. Sedangkan MySQL adalah sebuah sistem manajemen *database* yang bersifat open source yang dibuat dan dikembangkan oleh MySQL AB yang berada di Swedia.

MySQL dapat digunakan untuk membuat dan mengelola *database* beserta isinya. MySQL dapat dimanfaatkan untuk menambahkan, mengubah dan menghapus data yang berada dalam *database*, Selain itu merupakan sistem manajemen *database* yang bersifat *relational*. Artinya merupakan sistem manajemen *database* yang akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat (Bunafit Nugroho, 2004).

Menurut Abdul Kadir (2008:2) MySQL merupakan *software* yang tergolong sebagai DBMS yang bersifat *open source*. *Open source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL). Selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi dan bisa diperoleh secara gratis dengan mendownload di *internet*. MySQL awalnya dibuat oleh perusahaan konsultan bernama TeX yang berlokasi di Swedia. Saat ini pengembangan MySQL berada dibawah naungan MySQL AB. Sebagai *software* DBMS, MySQL memiliki sejumlah fitur seperti yang akan dijelaskan dibawah ini.

a. *Mutliplatform*

MySQL tersedia pada beberapa *platform* (*windows, linux, unix, dan lain-lain*)

b. Andal, cepat dan mudah digunakan.

MySQL tergolong sebagai *database server* (*server* yang melayani permintaan terhadap *database*) yang handal, dapat menangani *database* yang besar dengan kecepatan tinggi. Mendukung banyak sekali fungsi untuk mengakses *database* dan sekaligus mudah untuk digunakan.

c. Jaminan keamanan akses

MySQL mendukung pengamanan *database* dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur *user* tertentu agar bisa mengakses data yang bersifat rahasia (misal gaji pegawai), sedangkan *user* lain tidak boleh sesuai dengan hak aksesnya.

d. Dukungan SQL

Sebagaimana diketahui SQL merupakan bahasa standar dalam pengaksesan *database* rasional. Pengetahuan akan SQL akan memudahkan siapapun untuk menggunakan MySQL. Menurut Bunafit Nugroho (2008 : 92) sebagai sebuah program penghasil *database*, MySQL tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna (*Interface*) yang mungkin berguna sebagai program aplikasi pengakses *database* yang dihasilkan. MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open windows* seperti *Visual Basic, Delphi, PHP* dan lainnya. DBMS yang menggunakan bahasa SQL :

- 1) MySQL
- 2) Oracle
- 3) SQL Server 97, 2000 dll

2.11 Pengertian PHP *MyAdmin*

Php *MyAdmin* adalah sebuah aplikasi *open source* yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan php *MyAdmin*, anda dapat membuat *database*, membuat tabel, menambah dan memperbaharui data dengan GUI dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual (Akhmad Sofyan, 2003).

Feature yang ada di dalam php *MyAdmin* adalah, sbb :

- 1) *Feature* menjalankan SQL secara manual.
- 2) Status Informasi.
- 3) Pengaturan hak akses *user*.
- 4) *Export* dan *Import database* dan *table*

2.12 Analisis dan Perancangan Berbasis Objek

Pemrograman berorientasi objek adalah suatu cara baru dalam berpikir serta berlogika dalam menghadapi masalah-masalah yang akan dicoba diatasi dengan bantuan komputer. Tidak seperti pendahulunya (pemrograman terstruktur), pemrograman berorientasi objek mencoba melihat permasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi tertentu (Adi Nugroho, 2002:1).

Dalam analisis dan perancangan metodologi berorientasi objek menggunakan diagram UML (*Use case diagram, Class Diagram, Activity Diagram, Statechart Diagram, Sequence Diagram, Component Diagram, Deployment Diagram*). UML adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (Martin Fowler, 2005).

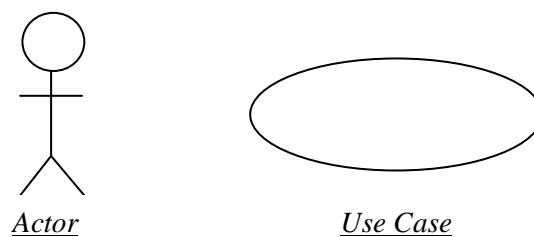
Kelebihan dari perancangan berorientasi objek menurut Britton dan Doake (2001) adalah sebagai berikut :

- a. *Maintainable* : pemeliharaan

- b. *Testable* : dicoba
- c. *Reuuseable* : dapat digunakan kembali
- d. *Able to cope with large and complex system* : dapat bekerjasama dengan sistem yang luas dan kompleks

2.12.1 Use Case Diagram

Use case diagram adalah teknik untuk merekam persyaratan fungsional sebuah sistem. *Use case* mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan (Martin Fowler, 2005:141). Diagram *Use case* dekat kaitannya dengan kejadian-kejadian. Kejadian (skenario) merupakan contoh apa yang terjadi ketika seseorang berinteraksi dengan sistem. *Use case* diagram dibuat untuk memvisualisasikan atau menggambarkan hubungan antara *actor* dan *use case*.



Gambar 2.2 *Use Case Diagram*

Actor mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. *Actor* hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*.

Sedangkan *use case* adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.

2.12.2 Class Diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class diagram* juga menunjukkan properti dan operasi sebuah class dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut (Martin Fowler, 2005:53).

Diagram kelas mempunyai 3 macam *relationship* (hubungan), sebagai berikut:

a. Association

Suatu hubungan antara bagian dari dua kelas, terjadi association antara dua kelas jika salah satu bagian dari kelas mengetahui yang lainnya dalam melakukan suatu kegiatan. Di dalam diagram, sebuah asosiasi adalah penghubung yang menghubungkan dua kelas.

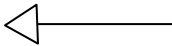

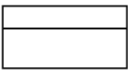


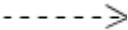
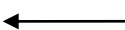
b. Aggregation

Suatu *association* dimana salah satu kelasnya merupakan bagian dari suatu kumpulan, *aggregation* memiliki titik pusat yang mencakup keseluruhan bagian, sebagai contoh: Order detail merupakan kumpulan dari order.

c. Generalization

Suatu hubungan turunan dengan mengasumsikan satu kelas merupakan suatu *super class* dari kelas yang lain. *Generalization* memiliki tingkatan yang berpusat pada *super class*, contoh: *Payment* adalah *super class* dari *Cash*, *Check*, *Credit*. Simbol –simbol yang digunakan pada *class diagram* ditunjukkan pada Tabel 2.1.






Tabel 2.1 Simbol Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

2.12.3 Activity Diagram

Activity Diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa hal, diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alir adalah diagram ini mendukung *behavior* paralel (Martin Fowler, 2005:163). Simbol –simbol yang digunakan pada *activity diagram* ditunjukkan pada Tabel 2.2.

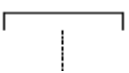

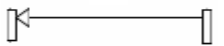
Tabel 2.2 Simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

2.12.4 Sequence Diagram

Sequence diagram adalah penjabaran *behavior* sebuah skenario tunggal. *Sequence diagram* menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek ini di dalam use case (Martin Fowler, 2005:81). *Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Simbol –simbol yang digunakan pada *sequence diagram* ditunjukkan pada Tabel 2.3.

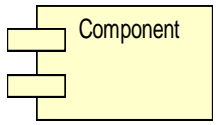

Tabel 2.3 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Menyatakan arah tujuan komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Response Message</i>	Menyatakan arah kembali dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

2.12.5 *Component Diagram*

Component merupakan bagian fisik dari sebuah sistem, karena menetap dikomputer tidak berada dianalisis. *Component* terhubung melalui antarmuka yang digunakan dan dibutuhkan (Martin Fowler, 2005:189). *Component* merupakan implementasi *software* dari sebuah atau lebih *class*. *Component* dapat berupa *sourcecode*, komponen biner, atau *executable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view*. Sehingga *component diagram* merepresentasikan dunia riil yaitu *component software* yang mengandung *component*, *Interface* dan *relationship*. Simbol –simbol yang digunakan pada *component diagram* ditunjukkan pada Tabel 2.4.

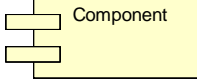
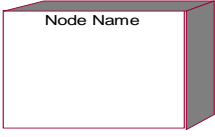

Tabel 2.4 Simbol-Simbol *Component Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Component	Sebuah komponen melambangkan sebuah entitas <i>software</i> dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.
2		Depedency	Sebuah <i>Dependency</i> digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus putus yang diarahkan kepada komponen.

2.12.6 *Deployment Diagram*

Deployment diagram menunjukkan susunan fisik sebuah sistem, menunjukkan bagian perangkat lunak mana yang berjalan pada perangkat keras (Martin Fowler, 2005:137). *Deployment diagram* juga menggambarkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Simbol –simbol yang digunakan pada *deployment diagram* ditunjukkan pada Tabel 2.5.

Tabel 2.5 Simbol-Simbol *Deployment Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Component	Pada <i>deployment diagram</i> , komponen-komponen yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka
2		Node	<i>Node</i> menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk <i>node</i> digambarkan sebagai sebuah kubus 3 dimensi.
3		Association	Sebuah <i>association</i> digambarkan sebagai sebuah garis yang menghubungkan dua <i>node</i> yang mengindikasikan jalur komunikasi antara element-elemen <i>hardware</i> .

2.13 *Webqual*

Webqual merupakan salah satu metode atau teknik pengukuran kualitas website berdasarkan persepsi pengguna akhir. Metode ini merupakan pengembangan dari *serqual* yang banyak digunakan sebelumnya pada pengukuran kualitas jasa. Instrumen penelitian pada *webqual* tersebut dikembangkan dengan metode *quality function development (QFD)*, yang bermakna: terstruktur dan proses disiplin yang memberikan makna untuk mengidentifikasi dan membawa suara customer melalui tiap-tiap tahap dari produk dan atau pengembangan pelayanan dan penerapannya (Barnes dan Vidgen:2002).

Webqual sudah mulai dikembangkan sejak tahun 1998 dan telah mengalami beberapa interaksi dalam penyusunan butir pertanyaan (Barnes dan Vidgen: 2002). *Webqual* disusun berdasarkan penelitian pada tiga area yaitu:

1. *Usability Quality*

Usability adalah mutu yang berhubungan dengan rancangan *site*.

2. *Information Quality*

Information adalah mutu dari isi yang terdapat pada *site*, pantas tidaknya informasi untuk tujuan pengguna.

3. *Interaction Quality*

Interaction Adalah mutu dari interaksi pelayanan yang dialami oleh pengguna ketika mereka menyelidiki kedalam *site* lebih dalam, yang terwujud dengan kepercayaan dan empati.

2.14 Konsep Basis Data

Menurut (Fathansyah, 2007:2), basis data atau *database* terdiri dari dua kata, yaitu Basis dan Data. Basis dikatakan sebagai markas atau gudang, tempat bersarang atau berkumpul, sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek yang direkam dalam bentuk angka, simbol, huruf, teks, gambar, bunyi dan kombinasinya. Sehingga Basis Data dapat didefinisikan dalam sejumlah sudut pandang sebagai berikut:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file, tabel, atau arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.15 Basis Data

Basis data adalah kumpulan dari item data yang saling berhubungan satu dengan lainnya, yang diorganisasikan berdasar sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan menggunakan *software* tertentu digunakan untuk melakukan manipulasi data untuk tujuan tertentu.

Suatu basis data adalah koleksi data yang bisa mencari secara menyeluruh dan sistematis memelihara informasi (Janner Simarmata, 2011). Basis data dapat diartikan sebagai himpunan kelompok data (arsip) yang saling berhubungan yang

diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah (Fatansyah, 2007).

2.16 Data

Data merupakan representasi fakta dunia nyata yang mewakili suatu objek yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, atau kombinasinya (Abdul Kadir, 2010). Tabel 2.6 menyajikan beberapa tipe data.

Tabel 2.6 Tipe Data

Tipe Data	Keterangan	Ukuran
<i>Varchar</i>	Tipe ini rangkaian dari karakter, memori yang dibutuhkan berdasarkan karakter yang digunakan ditambah 1 <i>byte</i> .	Maksimum terdiri atas 225 karakter
<i>Char</i>	Tipe ini rangkaian dari karakter, memori yang dibutuhkan bersifat statis	Maksimum terdiri atas 225 karakter
<i>Teks</i>	Tipe ini digunakan untuk menyatakan teks atau data yang bisa mengandung huruf, angka, dan karakter – karakter lain seperti tanda * dan &. Contoh pemakaiannya antara lain untuk nama orang, alamat, dan bahkan untuk angka yang tidak diaksudkan untuk di hitung (mis: no telpon).	Maksimum terdiri atas 65.535 karakter
Memo	Merupakan tipe data teks yang berukuran besar.	Dapat mencapai 16.777215 karakter
<i>Number</i>	Merupakan tipe data untuk suatu nilai bilangan yang bisa dihitung. Ada bermacam – macam pilihan lebih lanjut untuk nilai ini.	1,2,4,8, atau 16 <i>byte</i> tergantung detainya
<i>Data / Time</i>	Nilai jam dan tanggal dimulai dari tahun 100 sampai dengan 9999.	8 <i>byte</i>
<i>Currenncy</i>	Merupakan tipe data untuk nilai uang. Keakuratan sampai 15 digit disebelah kiri tanda pecahan dan 4 digit disebelah kanan tanda pecahan.	8 <i>byte</i>
<i>Yes / No</i>	Menyatakan data yang hanya memiliki dua kemungkinan saja. Contoh: besar salah,pria wanita.	1 <i>byte</i>
<i>Lookup wizard</i>	Memungkinkan berhubungan dengan nilai.	Biasanya 4 <i>byte</i>