

BAB II

TINJAUAN PUSTAKA

2.1 Sistem

Menurut Abdul Kadir (2014), Sistem adalah kumpulan elemen-elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Sebagai gambaran, jika sebuah sistem terdapat elemen yang tidak memberikan manfaat dalam mencapai tujuan yang sama maka elemen tersebut dapat dipastikan bukan bagian dari sistem. Sehingga dapat disimpulkan bahwa suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

2.2 Informasi

Menurut Abdul Kadir (2014) Informasi berarti data yang telah diproses sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut. Sebaliknya data merupakan sekumpulan baris fakta yang mewakili peristiwa yang terjadi pada organisasi atau lingkungan fisik sebelum diolah dalam suatu format yang dapat dipahami dan digunakan orang.

2.3 Pengertian Aplikasi

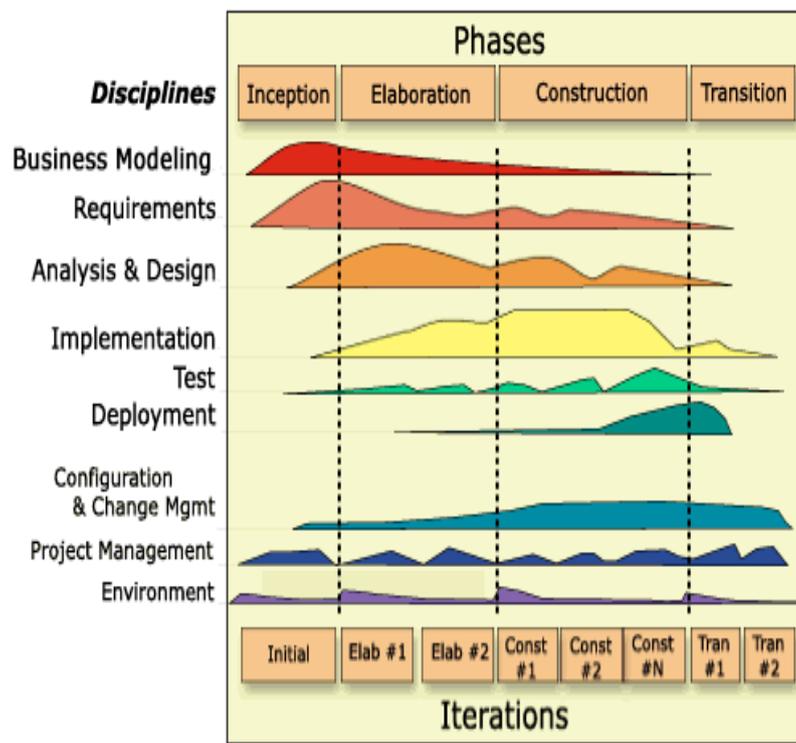
Aplikasi adalah satu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas. Misalnya termasuk perangkat lunak perusahaan, software akuntansi, perkantoran, grafis perangkat lunak dan pemutar media. Dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data. Program aplikasi merupakan program siap pakai. Program yang direka untuk melaksanakan suatu fungsi bagi pengguna atau aplikasi yang lain. Contoh-contoh aplikasiialah program pemroses kata dan *Web Browser*. Aplikasi akan menggunakan sistem operasi komputer dan aplikasi yang lainnya yang mendukung.

Klasifikasi aplikasi dapat dibagi menjadi 2 (dua) yaitu:

- a) Aplikasi *software* spesialis, program dengan dokumentasi tergabung yang dirancang untuk menjalankan tugas tertentu.
- b) Aplikasi paket, dengan dokumentasi tergabung yang dirancang untuk jenis masalah tertentu.

2.4 Rational Unified Process (RUP)

Sukamto dan Shalahuddin (2018:125), *RUP (Rational Unified Process)* adalah pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (*iterative*), fokus pada arsitektur (*architecture-centric*), lebih diarahkan berdasarkan penggunaan kasus (*use case driven*).



Gambar 2. 3 Metode RUP(Rasional Unified Process)

(Sumber: <http://www.ibm.com/developerworks/rational/library/feb05/krebs/krebsfig1.gif>)

2.5 Alat Analisis

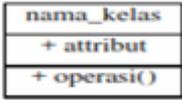
2.5.1 UML(*Unified Modeling Language*)

UML adalah sebuah kumpulan dari konvensi pemodelan yang digunakan, untuk menentukan atau menggambarkan sebuah *sistem software* yang terkait dengan objek.

2.5.2 *Class diagram*

Class diagram adalah gambar grafis mengenai struktur objek statis sebuah sistem menunjukkan kelas-kelas objek yang menyusun sebuah sistem dan juga hubungan antara kelas objek tersebut. Simbol *Class Diagram* dapat dilihat pada tabel di bawah ini :

Tabel 2.1 Simbol *Class Diagram*

No	Simbol	Keterangan
1	Kelas 	Kelas pada struktur sistem
2	Antarmuka / <i>interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
3	Asosiasi / <i>association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
4	Asosiasi berarah / <i>directed Association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

5	<p>Generalisasi</p> 	<p>Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus)</p>
6	<p>Kebergantungan / <i>dependency</i></p> 	<p>Relasi antarkelas dengan makna kebergantungan antarkelas</p>
7	<p>Agregasi / <i>aggregation</i></p> 	<p>Relasi antarkelas dengan makna semuabagian (<i>whole-part</i>)</p>

2.5.3 Use Case Diagram

Use Case Diagram adalah diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal atau pengguna. Secara grafis *Use Case Diagram* menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan interaksi dengan sistem.

Use case sendiri menggambarkan interaksi antara sistem, sistem eksternal, dan *user*, dengan kata lain mendeskripsikan siapa yang menggunakan sistem dan dengan cara apa pengguna mengharapkan interaksi dengan sistem itu. Simbol *Use Case diagram* dapat dilihat pada tabel di bawah ini

Tabel 2.2 Simbol Use Case Diagram

No	Simbol	Keterangan
1	<p><i>Use cases</i></p> 	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i></p>

2	<p>Aktor / <i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i></p>
3	<p>Asosiasi / <i>association</i></p> 	<p>Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor</p>
4	<p>Ekstensi / <i>extend</i></p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; ditambahkan, missal arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
5	<p>Generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya</p>

6	Menggunakan / <i>include</i> / <i>uses</i> <<include>> -----> <<uses>> ----->	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> .
---	--	--

2.5.4 Activity Diagram

Activity Diagram adalah diagram yang secara grafis menggambarkan aliran aktifitas baik proses bisnis atau *use case*. Diagram ini juga dapat digunakan untuk memodelkan aksi yang akan dilakukan saat sebuah operasi di eksekusi dan memodelkan hasil dari aksi tersebut. Simbol activity diagram dapat dilihat pada table di bawah ini

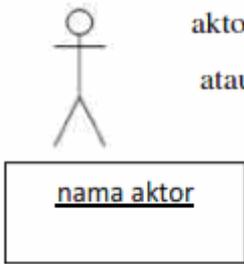
Tabel 2.3 Simbol Activity Diagram

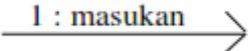
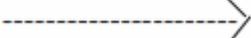
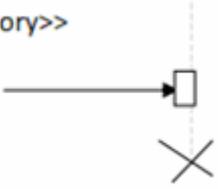
No	Simbol	Keterangan
1	Status Awal ●	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2	Aktivitas ▭	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3	Percabangan / <i>decision</i> ◇	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
4	Penggabungan / <i>join</i> —	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5	Status akhir ◎	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir

2.5.5 Sequence diagram

Sequence diagram menggambarkan bagaimana objek berinteraksi satu sama lain melalui pesan dalam *use case* atau suatu operasi. *Sequence* diagram menggambarkan bagaimana pesan dikirim dan diterima antara objek dan dalam satu urutan. Sebuah *sequence* diagram memiliki aktor beserta dengan pesan dan fungsi dimana pesan tersebut dikerjakan dengan berurutan

Tabel 2.4 Simbol Sequence Diagram

No	Simbol	Keterangan
1	Aktor  aktor atau <u>nama aktor</u>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama <i>actor</i>
2	Garis hidup / <i>lifeline</i> 	Menyatakan kehidupan suatu objek
3	Objek  nama objek : nama kelas	Menyatakan objek yang berinteraksi Pesan

4	<p style="text-align: center;">Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.</p>
5	<p>Pesan tipe <i>create</i></p> <p style="text-align: center;"><<create>></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
6	<p>Pesan tipe <i>call</i></p> <p>1 : nama_metode()</p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>
7	<p>Pesan tipe <i>send</i></p> <p>1 : masukan</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
8	<p>Pesan tipe <i>return</i></p> <p>1 : keluaran</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
9	<p>Pesan tipe <i>destroy</i></p> <p style="text-align: center;"><<destory>></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i></p>

2.6 Kamus Data (*Data Dictionary*)

Jogiyanto H.M, (2015). Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem. Simbol-simbol yang ada dalam kamus adalah sebagai berikut:

Simbol	Keterangan
=	Artinya adalah terdiri atas, diuraikan menjadi,
+	Artinya adalah dan
()	Artinya adalah opsional (pilihan Boleh ada atau tidak)
[]	Artinya adalah memilih salah satu alternatif, seleksi
{ }	Artinya adalah pengulangan
**	Artinya adalah komentar
@	Artinya adalah identifikasi atribut kunci
	Artinya adalah pemisahan alternatif []

Gambar 2.1 Simbol Kamus Data.

2.7 Gambaran Software yang Digunakan

2.7.1 Adobe Dreamweaver

Adobe Dreamweaver adalah salah satu program editor halaman web atau merupakan program penyunting-pengolah halaman web keluaran *Adobe Systems* yang dulu dikenal sebagai *Macromedia Dreamweaver* keluaran Macromedia. Program ini banyak digunakan oleh pengembang web karena fitur-fiturnya yang menarik dan kemudahan penggunaannya. Versi terakhir *Macromedia Dreamweaver* sebelum Macromedia dibeli oleh *Adobe Systems* yaitu versi 8. Versi terakhir *Dreamweaver* keluaran *Adobe Systems* adalah *Adobe Dreamweaver CS5* yang ada dalam *Adobe Creative Suite 5*.

2.7.2 PHP

PHP adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini, PHP banyak dipakai untuk program situs web dinamis, contoh terkenal dari aplikasi PHP adalah forum (phpBB) dan MediaWiki (*software* di belakang Wikipedia). Menurut Anhar (2010), PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Postnuke, Xaraya, dan lain-lain.

PHP dibuat pertama kali oleh Rasmus Lerdorf, yang pada awalnya dibuat untuk menghitung jumlah pengunjung pada home pagenya. Awalnya PHP kependekan dari *personal home page* saat itu namanya masih *Form Interpreted*. Selanjutnya pembuat PHP merilis kode sumber (*open source*) ke khalayak umum sehingga banyak programmer yang tertarik untuk mengembangkan PHP. (Virgi, 2011: 10-11)

Script PHP disisipkan langsung dalam tubuh *file* HTML yang ditandai dengan *tag* pembuka dan penutup. Sebagaimana diketahui, HTML (*HyperText Markup Language*) adalah bahasa standar untuk membuat halaman-halaman *web*. Ada dua pasang *tag* PHP yang dapat digunakan yaitu `<?php...>` dan `<script language="php">...</script>`.

Kode-kode PHP ditulis diantara *tag* pembuka dan penutup seperti dalam contoh-contoh berikut:

```
HTML
<?php
Kode PHP di sini;?>
HTML
Dan
```

HTML

```
<script language="php">
```

kode PHP di sini;

```
</script>
```

HTML

Script PHP diawali dengan tag (<?) dan diakhiri dengan tag (?>). Setiap baris perintah / statement harus diakhiri dengan menggunakan tanda titik koma (;). Umumnya setiap statement dituliskan dalam satu baris. Script PHP merupakan script yang digunakan untuk menghasilkan halaman-halaman *web*. Cara penulisannya dibedakan menjadi *embedded* dan *non embedded script*.

2.7.3. XAMPP

XAMPP adalah perangkat lunak bebas yang mendukung banyak sistem operasi yang merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program Apache HTTP *Server*, MySQL database dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam *General Public License* (GNU) dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

XAMPP adalah kepanjangan yang masing-masing hurufnya adalah:

- X** : Program ini dapat dijalankan di banyak sistem operasi, seperti Windows, Linux, Mac OS, dan juga Solaris.
- A** : **Apache**, merupakan aplikasi *web server*. Tugas utama Apache adalah menghasilkan halaman *web* yang benar kepada *user* berdasarkan kode PHP yang dituliskan oleh pembuat *web*, maka dapat saja suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman *web* yang dihasilkan.

- M** : **MySQL**, merupakan aplikasi database *server*. Perkembangannya disebut SQL yang merupakan kepanjangan dari *Structure Query Language*. SQL merupakan bahasa terstruktur yang digunakan untuk mengolah database. MySQL dapat digunakan untuk membuat dan mengelola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah dan menghapus data yang berada dalam database.
- P** : **PHP**, bahasa pemrograman *web*. Bahasa pemrograman PHP merupakan bahasa pemrograman untuk membuat *web* yang bersifat *server-side scripting*. PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. Sistem manajemen basis data yang sering digunakan bersama PHP adalah MySQL.
- P** : **Perl** adalah bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin Unix. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows, Power PC, BeOS, VMS, EBCDIC, dan PocketPC. (Rachmad Hakim, 2010: 120-121)

2.7.4 MySQL

MySQL merupakan sebuah basis data yang mengandung satu atau beberapa kolom. Tabel terdiri atas sejumlah basis dan setiap baris mengandung satu atau beberapa kolom. Didalam PHP telah menyediakan fungsi untuk koneksi ke basis data dengan sejumlah fungsi untuk pengaturan baik menghubungkan maupun memutuskan koneksi *server* database MySQL sebagai sarana untuk mengumpulkan informasi. (Yeni Kustiyahningsih, Devie Rosa Anamisa, 2010: 145-146).

Database adalah sistem penyimpanan beragam jenis data dalam sebuah entitas yang besar untuk diolah sedemikian rupa agar mudah dipergunakan lagi. Data yang disimpan bisa sangat variatif (angka, teks, gambar, suara, dan jenis data multi-media lainnya). Basis data merupakan kumpulan dari data yang saling

berhubungan satu dengan yang lainnya, tersimpan di perangkat keras computer dan digunakan perangkat lunak untuk memanipulasinya. Database merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai. (Sucipto, 2012: 137).

MySQL adalah sistem manajemen basisdata relasi yang bersifat terbuka atau *open source*. Sistem manajemen basisdata ini adalah hasil pemikiran dari Michael “Monty” Widenius, David Axmark, dan Allan Larson pada tahun 1995. Tujuan awal ditulisnya program MySQL adalah untuk mengembangkan aplikasi *web*. MySQL menggunakan bahasa standar SQL (*Structure Query Language*) sebagai bahasa interaktif dalam mengelola data.

Perintah SQL sering juga disebut *Query*. MySQL menawarkan berbagai keunggulan dibandingkan *database server* lain.

Berikut ini adalah beberapa keunggulan MySQL:

1. Mampu menangani jutaan *user* dalam waktu yang bersamaan.
2. Mampu menampung lebih dari 50.000.000 *record*.
3. Sangat cepat mengeksekusi perintah.
4. Memiliki *user privilege system* yang mudah dan efisien.

Kelemahan MySQL:

1. Untuk koneksi ke bahasa pemrograman visual seperti vb, delphi, dan foxpro, mysql kurang support, karena koneksi ini menyebabkan field yang dibaca harus sesuai dengan koneksi dari program visual tersebut, dan ini yang menyebabkan mysql jarang dipakai dalam program visual.
2. Data yang ditangani belum begitu besar.