

BAB II

LANDASAN TEORI

2.1 Pemrograman Berorientasi Objek

Berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya [1].

Suatu program disebut dengan pemrograman berbasis objek (OOP) karena terdapat :

1. *Encapsulation* (pembungkusan)
 1. Variabel dan *method* dalam suatu obyek dibungkus agar terlindungi
 2. Untuk mengakses, variabel dan *method* yang sudah dibungkus tadi perlu *interface*
 3. Setelah variabel dan *method* dibungkus, hak akses terhadapnya bisa ditentukan.
 4. Konsep pembungkusan ini pada dasarnya merupakan perluasan dari tipe dan struktur

2. *Inheritance* (pewarisan)
 - a. Sebuah *Class* bisa mewariskan atribut dan metodenya ke *Class* yang lain
 - b. *Class* yang mewarisi disebut *superClass*
 - c. *Class* yang diberi warisan disebut *subClass*
 - d. Sebuah *subClass* bisa mewariskan atau berlaku sebagai *superClass* bagi *Class* yang lain -> disebut *multilevel inheritance*
 - e. *SubClass* memiliki atribut dan *method* yang spesifik yang membedakannya dengan *superClass*, meskipun keduanya mirip (dalam hal kesamaan atribut dan *method*).
 - f. Pada pembuatan *subClass*, programmer bisa menggunakan ulang *source code* dari *superClass* yang ada -> ini yang disebut dengan istilah *reuse*.
 - g. *Class-Class* yang didefinisikan dengan atribut dan *method* yang bersifat umum yang berlaku baik pada *superClass* maupun *subClass* disebut dengan *abstract Class*.

3. *Polymorphism* (perbedaan bentuk)

Polimorfisme artinya penyamaran dimana suatu bentuk dapat memiliki lebih dari satu bentuk.

2.2 Pengertian Aplikasi

Pengertian aplikasi menurut para ahli adalah sebagai berikut:[16]

1. Menurut Jogiyanto adalah penggunaan dalam suatu komputer, instruksi (instruction) atau pernyataan (statement) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.
2. Menurut Kamus Kamus Besar Bahasa Indonesia adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.
3. Menurut Rachmad Hakim S, adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya.
4. Menurut Harip Santoso, adalah suatu kelompok file (*form, class, report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi payroll, aplikasi fixed asset.

2.3 Pengertian Layanan

Menurut Barata bahwa suatu pelayanan akan terbentuk karena adanya proses pemberian layanan tertentu dari pihak penyedia layanan kepada pihak yang dilayani.³ Ada beberapa ciri pelayanan yang baik yang dapat memberikan kepuasan kepada pelanggan :

- a. Memiliki karyawan yang profesional khususnya yang berhadapan langsung dengan pelanggan.
- b. Tersedianya sarana dan prasaranayang baik yang dapat menunjang kelancaran produk ke pelenggan secara cepat dan tepat.

- c. Tersedianya ragam produk yang diinginkan. Dalam artian konsumen sekali berhenti dapat membeli beragam produk dengan kualitas produk dan pelayanan yang mereka inginkan.
- d. Bertanggung jawab kepada setiap pelanggan dari awal hingga selesai.
- e. Mampu melayani secara cepat dan tepat, tentunya jika dibandingkan dengan pihak pesaing.
- f. Mampu berkomunikasi dengan jelas, menyenangkan dan mampu menangkap keinginan dan kebutuhan pelanggan.

2.4 Pengertian Akademik

Kata akademik berasal dari bahasa Yunani yakni *academos* yang berarti sebuah taman umum (*plasa*) di sebelah barat laut kota Athena. Nama *Academos* adalah nama seorang pahlawan yang terbunuh pada saat perang legendaris Troya. Pada *plasa* inilah filosofi Socrates berpidato dan membuka arena perdebatan tentang berbagai hal. Tempat ini juga menjadi tempat Plato melakukan dialog dan mengajarkan pikiran-pikiran filosofinya kepada orang-orang yang datang. Sesudah itu, kata *academos* berubah menjadi akademik, yaitu semacam tempat perguruan. Para pengikut perguruan tersebut disebut *academist*, sedangkan perguruan semacam itu disebut *academia*. Berdasarkan hal ini, inti dari pengertian akademik adalah keadaan orang-orang bisa menyampaikan dan menerima gagasan, pemikiran, ilmu pengetahuan, dan sekaligus dapat mengujinya secara jujur, terbuka, dan leluasa [13].

2.5 Pengertian Internet

Internet merupakan jaringan global komputer dunia, besar dan sangat luas sekali dimana setiap komputer saling terhubung satu sama lainnya dari negara ke negara lainnya di seluruh dunia dan berisi berbagai macam informasi, mulai dari text, gambar, audio, video, dan lainnya. *Internet* itu sendiri berasal dari kata *Interconnection Networking*, yang berarti hubungan dari banyak jaringan komputer dengan berbagai tipe dan jenis, dengan menggunakan tipe komunikasi seperti telepon, satelit, dan lainnya.

Dalam mengatur integrasi dan komunikasi jaringan komputer ini menggunakan protokol yaitu TCP/IP. TCP (*Transmission Control Protocol*) bertugas untuk memastikan bahwa semua hubungan bekerja dengan benar, sedangkan IP (*Internet Protocol*) yang mentransmisikan data dari satu komputer ke komputer lain. TPC/IP secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat digunakan, mengatur dan mengirimkan paket - paket pengiriman data. Untuk dapat ikut serta menggunakan fasilitas *Internet*, Anda harus berlangganan ke salah satu ISP (*Internet Service Provider*) yang ada dan melayani daerah Anda. ISP ini biasanya disebut penyelenggara jasa *Internet*. Anda bisa menggunakan fasilitas dari Telkom seperti Telkomnet Instan, speedy dan sebagainya[5].

Internet memberikan banyak sekali manfaat, ada yang bisa memberikan manfaat baik dan buruk. *Internet* ini memungkinkan pengguna komputer di seluruh dunia untuk saling berkomunikasi dan berbagi informasi dengan cara saling mengirimkan *email*, menghubungkan komputer satu ke komputer yang lain, mengirim dan menerima *file* dalam bentuk text, audio, video, membahas topik tertentu pada *newsgroup*, *website social networking* dan lain-lain.

Berikut adalah beberapa pengertian tentang istilah yang ada dalam *internet*:

1. *Network* (jaringan) adalah sistem yang saling terhubung dari berbagai komputer, terminal dan saluran serta peralatan komunikasi dengan menggunakan jalur transmisi.
2. WWW (*World Wide Web*) atau yang sering disebut sebagai "*web*" merupakan sebuah sistem dimana informasi dalam bentuk teks, gambar, suara, dan lain - lain yang tersimpan dalam sebuah *Internet webserver* dipresentasikan dalam bentuk *hypertext*. Informasi di web dalam bentuk teks umumnya ditulis dalam format HTML (*Hypertext Markup Language*). Informasi lainnya disajikan dalam bentuk grafis atau gambar (dalam format GIF, JPG, PNG), suara (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World).
3. *Web server* adalah *software* yang menjadi tulang belakang dari *world wide web(www)*. *Web server* menunggu permintaan dari client yang menggunakan *browser*. Jika ada permintaan dari *browser*, maka *web server*

akan memproses permintaan itu kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke *browser*. *Web server*, untuk berkomunikasi dengan *client*-nya (*web browser*) mempunyai protokol sendiri, yaitu HTTP (*hypertext transfer protocol*). Dengan protokol ini, komunikasi antar *web server* dengan *client*-nya dapat saling dimengerti dan lebih mudah.

4. HTTP (*hypertext transfer protocol*) potokol yang memungkinkan bermacam-macam komputer saling berkomunikasi dengan menggunakan bahasa HTML. *Hypertext* mempunyai arti bahwa seorang pengguna *Internet* dengan *web browser*-nya dapat membuka dan membaca dokumen-dokumen yang ada dalam komputernya atau bahkan jauh tempatnya sekalipun.
5. *Browser* adalah suatu perangkat lunak yang digunakan untuk dapat menjelajah *Internet*. Perangkat lunak ini diperlukan untuk menampilkan halaman suatu *website*. *Web browser* adalah paket software yang menyediakan *interface* pemakai untuk mengakses situs *Web Internet*, *Intranet*, dan *Eksternet*.
6. *Website* atau biasa juga disebut situs *web* adalah sebuah tempat di *Internet* tempat beradanya suatu kumpulan halaman *web* (*webpage*). Tempat ini biasanya ditandai dengan sebuah alamat *website*.
7. *Home page* adalah suatu halaman depan dari sebuah *website*. Ketika pertama kali mengakses suatu alamat *website* maka biasanya akan langsung menuju ke halaman depan (*home page*) *website* tersebut.

2.6 Apache

Pada awal mulanya, Apache merupakan perangkat lunak sumber terbuka yang menjadi *alternatif* dari *server web* Netscape (sekarang dikenal sebagai *Sun Java System Web Server*). Asal mula nama Apache berasal ketika sebuah *server web* populer yang dikembangkan pada awal 1995 yang bernama NCSA HTTPd 1.3 memiliki sejumlah perubahan besar terhadap kode sumbernya (*patch*). Tetapi pada halaman FAQ situs *web* resminya, disebutkan bahwa "Apache" dipilih untuk menghormati suku asli Indian Amerika Apache (Indé), yang dikenal karena keahlian dan strategi perangnya. Versi 2 dari Apache ditulis dari awal tanpa mengandung kode sumber dari NCSA.

Server HTTP Apache atau *Server Web/WWW Apache* adalah *server web* yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft Windows, dan Novell Netware serta *platform* lainnya) yang berguna untuk melayani dan memfungsikan situs *web*. Ada banyak *web server* yang berkembang dan sering digunakan dalam membangun aplikasi berbasis *web*, seperti PWS dan IIS yang dipakai oleh ASPnya Microsoft Web Server Netscape, Qitami, Caudium, dan Apache[7].

Berdasarkan *survey* Netcraft.com, Apache merupakan *web server* yang paling populer digunakan di dunia. Beberapa kelebihan Apache antara lain:

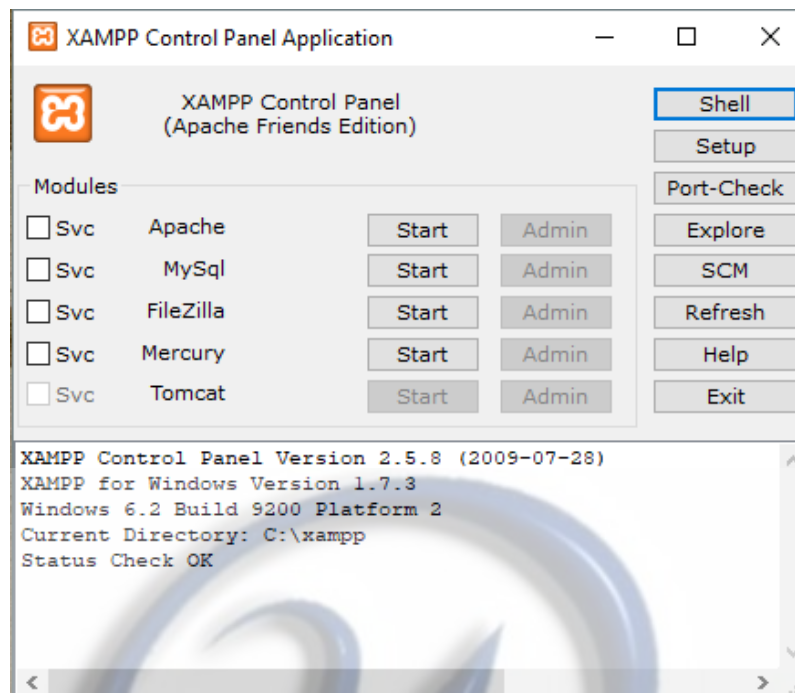
1. Free of charge, berarti tidak harus membayar *lisensi* kepada pembuat untuk menggunakannya.
1. Dapat diakses (API ke berbagai *scripting language*) dan digabung dengan berbagai aplikasi lain (*database server, SSL, ext*) dan sebagainya.
2. Waktu pemrosesan lebih cepat dan tangguh dengan konfigurasi yang benar.
2. Dapat dilakukan *setting* dan *instalasi* sesuai dengan kebutuhan dengan adanya *modules* dan DSO-nya.
3. Memiliki kemampuan *Advanced Setting* dan *Configuration Support*.

2.7 CodeIgniter

CodeIgniter adalah sebuah *framework* PHP yang dapat mempercepat untuk membuat sebuah aplikasi web. Dilengkapi banyak *library* dan *helper* yang berguna di dalamnya dan tentunya mempermudah proses *development*. Sedangkan Twitter Bootstrap adalah sebuah alat bantu (*framework*) HTML dan CSS untuk membuat sebuah tampilan halaman website yang elegant dan *support* segala macam device. Dengan implementasi Twitter Bootstrap pada *framework* CodeIgniter akan mempermudah dan mempercepat pembuatan maupun pengembangan website dinamis tanpa harus kesulitan membuat desain [12].

2.8 XAMPP

XAMPP adalah salah satu pake *software* web server yang terdiri dari Apache, MySQL, PHP, dan PHPMyAdmin. Proses instalasi XAMPP sangat mudah, karena tidak perlu melakukan konfigurasi Apache, PHP dan MySQL secara manual, XAMPP melakukan instalasi dan konfigurasi secara otomatis [17].



Gambar 2.1. Tampilan Control Panel XAMPP[17]

2.9 Framework

Framework aplikasi adalah desain dan program kerangka aplikasi yang digunakan untuk membangun aplikasi-aplikasi yang sejenis. *Framework* memiliki banyak pustaka tingkat tinggi yang dapat digunakan secara berulang. Umumnya *framework* perangkat lunak didesain secara berorientasi objek sehingga desain dan programnya tersedia sebagai class-class abstrak. *Framework* aplikasi mendefinisikan arsitektur aplikasi sehingga penambahan fungsi-fungsi lain dapat dilakukan dengan menambah modul-modul tanpa perlu memodifikasi kode program yang sudah ada, kecuali file-file konfigurasi [10].

2.10 Personal Home Page (PHP)

PHP adalah singkatan dari "PHP Hypertext Preprocessor", yang merupakan sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, Java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat.

Kelebihan PHP yang paling utama adalah pada konektivitasnya dengan *system database* di dalam web. Sistem *database* yang dapat didukung oleh PHP

adalah : Oracle, MySQL, Sybase, PostgreSQL, dan lainnya. PHP dapat berjalan di berbagai sistem operasi seperti windows 98/NT, UNIX/LINUX, solaris maupun macintosh. PHP merupakan *software yang open source* yang dapat anda *download* secara gratis. *Software* ini juga dapat berjalan pada *web server* seperti PWS (*Personal Web Server*), Apache, IIS, AOLServer, fhttpd, phttpd dan sebagainya. PHP juga merupakan bahasa pemrograman yang dapat kita kembangkan sendiri seperti untuk menambah fungsifungsi baru. PHP mendukung komunikasi dengan layanan seperti protocol IMAP, SNMP, NNTP, POP3 dan bahkan HTTP. PHP dapat diinstal sebagai bagian atau modul dari apache *web server* atau sebagai *CGI script* yang mandiri. Banyak keuntungan yang dapat diperoleh jika menggunakan PHP sebagai modul dari apache, di antaranya adalah : tingkat keamanan yang cukup tinggi, waktu eksekusi yang lebih cepat dibandingkan dengan bahasa pemrograman *web* lainnya yang berorientasi pada *server-side scripting*, akses ke *system database* yang lebih fleksibel seperti MySQL[15].

Skrip PHP berkedudukan sebagai *tag* dalam bahasa HTML. Adapun kode berikut adalah contoh kode PHP yang berada di dalam kode HTML:

```
<HTML>
  <HEAD>
    <TITLE>Latihan Pertama</TITLE>
  </HEAD>
  <BODY>
    Selamat Belajar PHP, <BR>
    <?php
      Printf ("Tgl. Sekarang: %s ", Date("d F Y"));
    ?>
  </BODY>
</HTML>
```

Kode di atas disimpan dengan ekstensi *.php*. Kode PHP 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.

Konsep kerja PHP pada prinsipnya serupa dengan kode HTML, hanya saja ketika berkas PHP yang diminta didapatkan oleh *web server*, isinya segera dikirimkan ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke *web server*. Selanjutnya, *web server* menyampaikan ke klien[7].

2.11 *My Structured Query Language (MySQL)*

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*database management system*) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial[9].

MySQL sebenarnya merupakan turunan salah satu konsep utama dalam *database* sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem *database* (DBMS) dapat diketahui dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan *database server* lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh *single user*, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan *Interbase*. MySQL juga memiliki beberapa keistimewaan yang lain, antara lain :

1. *Portabilitas*. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan lain-lain.
- b. *Open Source*. MySQL didistribusikan secara *open source*, dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.
- c. *Multiuser*. MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
- d. *Performance tuning*. MySQL memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
- e. *Jenis Kolom*. MySQL memiliki tipe kolom yang sangat kompleks, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
- f. *Perintah dan Fungsi*. MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
- g. *Keamanan*. MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
- h. *Skalabilitas dan Pembatasan*. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

2.12 Cascading Style Sheet (CSS)

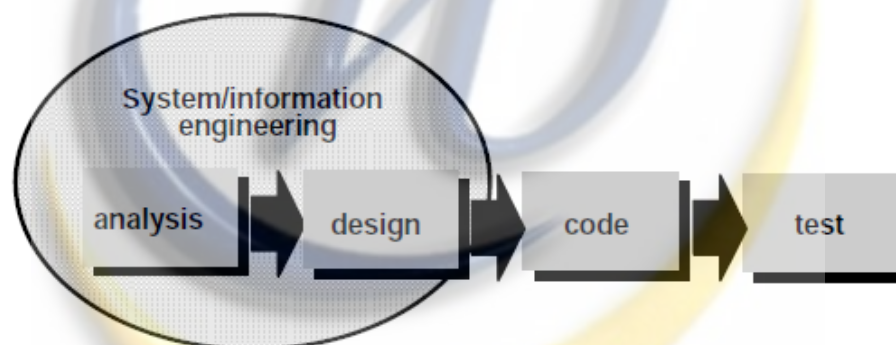
CSS merupakan kependekan dari *Cascading Style Sheet* yang berfungsi untuk mengatur tampilan dengan kemampuan jauh lebih baik dari tag maupun atribut standar HTML (*Hypertext Markup Language*). CSS sebenarnya adalah suatu kumpulan atribut untuk fungsi format tampilan dan dapat digunakan untuk mengontrol tampilan banyak dokumen secara bersamaan. Keuntungan menggunakan CSS yaitu jika ingin mengubah format dokumen, maka tidak perlu mengedit satu persatu.

Penggunaan CSS ada 2 (dua) cara yaitu dengan menyisipkan kode CSS langsung dalam kode HTML atau simpan menjadi file tersendiri berekstensi .css. Dengan

menyimpan sebagai file tersendiri akan lebih memudahkan untuk mengontrol tampilan dalam banyak dokumen secara langsung. CSS mendapat dukungan penuh pada browser versi 4 (empat) dan pada versi sebelumnya, hanya dengan Internet Explorer yang masih mampu mengenali CSS. Perlu diketahui bahwa tampilan CSS dapat berbeda bila ditampilkan pada menu browser yang berbeda[10].

2.13 Metode *Waterfall*

Model pengembangan yang digunakan pada penelitian ini ialah model *Waterfall*. Model *Waterfall* merupakan model pengembangan sistem informasi yang sistematis dan sekuensial [10]. Alasan penggunaan model *Waterfall* sebagai metode pengembangan Aplikasi Layanan Akademik ialah kebutuhan pihak prodi telah terdefinisi secara jelas dan tahap-tahap pada model *Waterfall* terstruktur secara jelas. Tahapan pengembangan sistem pada model *Waterfall* dapat diilustrasikan pada gambar berikut :



Gambar 2.2. Kerangka Kerja Pengembangan Sistem *Waterfall*[25]

1. Analysis

Pada tahap ini, merupakan proses analisa kebutuhan sistem. Pengembang mengumpulkan data-data sebagai bahan pengembangan sistem. Pengumpulan data dapat dilakukan dengan teknik wawancara, teknik observasi, dan teknik kuisioner [22].

2. Design

Proses desain adalah proses multi langkah yang berfokus pada empat atribut, yaitu : struktur data, arsitektur perangkat lunak, representasi interface, dan detail prosedural. Proses desain menterjemahkan hasil analisis ke dalam representasi perangkat lunak.

3. Code

Pada tahap ini desain diterjemahkan ke dalam program perangkat lunak. Pada tahap pengimplementasian ke dalam kode program akan bergantung pada hasil desain perangkat lunak pada tahap sebelumnya.

4. Test

Setelah pengkodean, dilanjutkan dengan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan untuk mengetahui kesesuaian hasil output dari sistem dengan kebutuhan yang telah dirancang pada tahap analisis.

2.14 Unified Modeling Language (UML)

Unified modeling language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal yang membantu pendeskripsian dan desain 15 sistem perangkat lunak khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek. Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corps. UML menyediakan notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan [19].

2.13 Use Case Diagram

Use Case adalah deskripsi fungsi sistem dari sudut pandang pengguna. *Use Case* diagram digunakan untuk menunjukkan fungsi sistem yang akan disediakan dan menunjukkan pengguna mana yang akan berkomunikasi dengan sistem. *Use Case* diagram ini dikembangkan oleh Jacobson et al. (1992) *Use Case* diagram menunjukkan tiga aspek dalam sistem [23] yaitu:

1. Actor

Actor mewakili peran yang dimiliki orang, sistem lain, atau *device* ketika berkomunikasi dengan *Use Case* tertentu dalam sistem dimana satu *actor* dapat mewakili beberapa orang atau pekerjaan.

2. Use Case




Sebuah *Use Case* menggambarkan sebuah fungsi yang dilakukan oleh sistem untuk mencapai tujuan pengguna dimana digambarkan dengan bentuk *eclipse*.


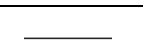
3. Batasan sistem atau subsistem

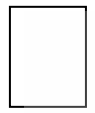



Batasan yang mengelilingi fungsi-fungsi dalam sistem atau *subsistem* dimana berbentuk persegi panjang.

Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain diluarnya.

Tabel 2.1. Deskripsi Bagian-bagian *Use Case* Diagram[22]

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<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>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>Use Case</i> sumber secara <i>eksplisit</i> .

5		<i>Extend</i>	Menspesifikasikan bahwa <i>Use Case</i> target memperluas perilaku dari <i>Use Case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.







No	Gambar	Nama	Keterangan
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.14 Activity Diagram

Activity diagram menggambarkan berbagai alir aktifitas dalam sebuah sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram tidak menggambarkan sifat internal dari sebuah sistem dan interaksi antara beberapa sub sistem secara eksak, tetapi lebih menggambarkan proses dan jalur-jalur aktivitas dari level atas secara umum [22].



Tabel 2.2. Deskripsi Bagian-bagian *Activity Diagram*[22]

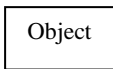
Simbol	Keterangan
	Start Point
	End Point
	Activities
	Fork (Percabangan)
	Join (Penggabungan)
	Decision
Swimlane	Sebuah cara untuk mengelompokkan activity berdasarkan Actor (mengelompokkan activity dalam sebuah urutan yang sama)


2.15 *Sequence Diagram*

Menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem[19].

Tabel 2.3. Deskripsi Bagian-bagian *Sequence Diagram*[19]

No	Gambar	Nama	Keterangan
1		<i>Actor</i>	<i>Actor</i> adalah pesan dari seseorang atau sistem lain yang bertukar informasi dengan sistem yang lainnya, kemudian lifeline berhenti atau mulai pada titik yang tepat.
2		<i>Message</i>	<i>Message</i> adalah komunikasi antar objek yang membawa informasi dan hasil pada sebuah aksi.

3		<i>Object Lifeline</i>	<i>Object Lifeline</i> menunjukkan keberadaan dari sebuah objek terhadap waktu. Yaitu objek dibuat atau dihilangkan selama suatu periode waktu diagram ditampilkan, kemudian lifeline berhenti atau mulai pada titik yang tepat.
---	---	------------------------	--

No	Gambar	Nama	Keterangan
4		<i>Activation</i>	<i>Activation</i> menampilkan periode waktu selama sebuah objek atau aktor melakukan aksi.

2.16 Class Diagram

Class diagram adalah sebuah diagram yang menunjukkan hubungan antar *Class* yang didalamnya terdapat atribut dan fungsi dari suatu objek.

Class diagram menggambarkan struktur dan deskripsi *Class*, package dan beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain [22].

Class memiliki tiga area pokok :

1. Nama
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki salah satu sifat berikut :

1. Private, tidak dapat dipanggil dari luar *Class* yang bersangkutan.
2. Protected, hanya dapat dipanggil oleh *Class* yang bersangkutan dan anak-anak yang mewarisinya.
3. Public, dapat dipanggil oleh siapa saja.

Class diagram mempunyai 3 relasi dalam penggunaannya, yaitu :

1. Assosiation

Assosiation adalah sebuah hubungan yang menunjukkan adanya interaksi antar *Class*. Hubungan ini dapat ditunjukkan dengan garis dengan mata panah terbuka di ujungnya yang mengindikasikan adanya aliran pesan dalam satu arah.

2. Generalization

Generalization adalah sebuah hubungan antar *Class* yang bersifat dari khusus ke umum.

3. Constraint

Constraint adalah sebuah hubungan yang digunakan dalam sistem untuk memberi batasan pada sistem sehingga didapat aspek yang tidak fungsional.

