

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam mendukung penelitian ini digunakan beberapa landasan teori bersumber dari beberapa jurnal dan berkaitan dengan pokok bahasan sebagai berikut :

2.1.1 Perancangan Sistem Informasi Tata Tertib Siswa Pada Smp Negeri 1 Jepara Dengan Menggunakan Sms Gateway

(Dafit Nur Hidayanto, 2013) Sekolah adalah salah satu lembaga yang bertugas membentuk kepribadian siswa agar mempunyai kepribadian yang luhur mulia serta disiplin yang tinggi. Sekolah Menengah Pertama sebagai salah satu pendidikan formal merupakan Sekolah yang sangat berpengaruh terhadap pembentukan pribadi siswa. Dalam kenyataan sehari – hari banyak dijumpai siswa yang tidak disiplin dan menyimpang dari norma. Untuk membentuk kepribadian siswa yang luhur mulia serta disiplin yang tinggi diperlukan peran sekolah serta orang tua. Dalam era komputerisasi saat ini dibutuhkan aplikasi untuk mempercepat proses informasi antara sekolah dan orang tua dalam hal ketertiban siswa sehingga orang tua dapat bekerja sama dengan pihak sekolah untuk membantu menangani perilaku siswa. Dengan adanya sistem informasi tata tertib siswa pada SMP Negeri 1 Jepara diharapkan orang tua mengetahui pelanggaran – pelanggaran anaknya secara cepat sehingga dapat memperkecil tingkat kenakalan siswa disekolah.

2.1.2 Sistem Pemantauan Siswa Sekolah Berbasis Web

(Budi Utomo, 2012) Lembaga pendidikan sebagai suatu wadah yang berfungsi sebagai tempat melaksanakan proses belajar – mengajar, pelatihan dan pengembangan terhadap anak didiknya, di era globalisasi ini diharapkan bisa mengikuti perkembangan ilmu pengetahuan dalam bidang sistem pemantauan untuk menunjang kegiatan akademik secara cepat, tepat, akurat dan berkualitas.

Pemantauan kegiatan akademik merupakan kegiatan utama di dunia pendidikan. Pondok Pesantren Darul Muttaqien sebagai salah satu instansi pendidikan tentunya harus melaksanakan kegiatan tersebut sebagai kegiatan wajib dalam pelaksanaan kegiatan belajar mengajar siswa. Akan tetapi, kegiatan tersebut belum dapat dilaksanakan secara optimal. Orang tua siswa masih mengalami kesulitan dalam memantau kegiatan belajar anak mereka di sekolah.

Pemberitahuan tentang prestasi (nilai) biasanya hanya dilakukan pada saat penerimaan raport sekolah. Orang tua siswa hanya bisa mendapatkan hasil akhir dari kegiatan belajar anaknya, tanpa bisa memantau proses akademik anak mereka. Pemberitahuan untuk siswa yang bermasalah dilakukan dengan cara mengirim surat dan terkadang surat tersebut tidak tersampaikan. Selain itu, penilaian dari guru juga masih dilakukan secara manual, sehingga orang tua ataupun siswa masih kesulitan mengetahui nilai pada saat proses belajar mengajar.

Sistem pemantauan berbasis web yang dibangun mampu memudahkan pihak sekolah dalam memantau dan mengontrol absensi dan informasi nilai akademik siswa secara mudah dan cepat tanpa harus membuka buku besar, serta memudahkan orang tua untuk memperoleh informasi yang berkaitan dengan akademik siswa disekolah. Sistem pemantauan siswa berbasis *website* pada

Pondok Pesantren Darul Muttaqien menggunakan *software notepad++* yang telah di instal sebelumnya. Dan tahap selanjutnya yaitu membuat *database* menggunakan *XAMPP (MySQL)* untuk menyimpan data – data siswa, dan tahap implementasi program aplikasi sistem dibuat menggunakan bahasa pemrograman *PHP*. Sehingga akan menghasilkan sistem pemantauan yang efektif dan efisien.

2.2 Tinjauan Studi

Dalam mendukung penelitian ini digunakan beberapa tinjauan studi yang sesuai dan berkaitan dengan pokok bahasan sebagai berikut :

2.2.1 Pengertian Sistem

(McLeod, 2010) dalam bukunya yang berjudul “*Management Information System*” adalah sekelompok elemen – elemen yang terintegrasi dengan maksud yang sama untuk mencapai tujuan.

(Satzinger, Jackson, dan Burd, 2010) sistem merupakan sekumpulan komponen yang saling berhubungan dan bekerja bersama untuk mencapai suatu tujuan. Berdasarkan kedua pengertian diatas, maka dapat disimpulkan bahwa pengertian sebuah sistem adalah sekumpulan elemen yang terintegasi dan bekerja bersama guna mencapai suatu tujuan tertentu.

2.2.2 Klasifikasi Sistem

(Yakub, 2012) pada buku “Pengantar Sistem Informasi”, Sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya :

- a. Sistem Abstrak (*Abstract System*) Sistem Abstrak adalah sistem yang berupa pemikiran atau ide – ide yang tidak tampak secara fisik. Sistem teologia yang berisi gagasan tentang hubungan manusia dengan Tuhan merupakan contoh *abstract system*.
- b. Sistem Fisik (*Physical System*) Sistem Fisik adalah sistem yang ada secara fisik, sistem komputer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system*.
- c. Sistem Tertentu (*Deterministic System*) Sistem Tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.

- d. Sistem Tak Tentu (*Probabilistic System*) Sistem Tak Tentu adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksikan dengan pasti.
- e. Sistem Tertutup (*Close System*) Sistem Tertutup merupakan sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan, misalnya reaksi kimia dalam tabung terisolasi.
- f. Sistem Terbuka (*Open System*) Sistem Terbuka adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan.

2.2.3 Pengertian Informasi

(McLeod kutip oleh Yakub, 2012) Informasi adalah data yang diolah menjadi bentuk lebih berguna dan lebih berarti bagi yang menerimanya.

(Tata Sutabri, 2012) Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Dari kedua pendapat diatas, dapat disimpulkan bahwa informasi adalah hasil dari prosedur organisasi berupa data yang telah diolah menjadi sebuah bentuk yang lebih berarti dan bermanfaat bagi penerimanya khususnya dalam mengambil suatu keputusan untuk mengendalikan organisasi.

2.2.4 Sistem Informasi

(Satzinger, Jackson, dan Burd, 2012) Sistem Informasi adalah kumpulan dari komponen – komponen yang mengumpulkan, memproses, menyimpan, dan menyediakan *output* dari setiap informasi yang dibutuhkan dalam proses bisnis serta aplikasi yang digunakan melalui perangkat lunak, *database* dan bahkan proses manual yang terkait.

(Gelinas dan Dull, 2012) Sistem Informasi merupakan sistem yang di buat secara umum berdasarkan seperangkat komputer dan komponen manual yang dapat dikumpulkan, disimpan dan diolah untuk menyediakan output kepada user. Dengan demikian dapat disimpulkan bahwa sistem informasi adalah suatu kombinasi modul yang terorganisir yang berasal dari komponen – komponen yang terkait dengan *hardware*, *software*, *people* dan *network* berdasarkan seperangkat komputer dan menghasilkan informasi untuk mencapai tujuan.

Dari kedua pendapat diatas, dapat disimpulkan bahwa Sistem informasi adalah suatu sekumpulan elemen atau prosedur dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi untuk pihak – pihak tertentu sesuai dengan kebutuhannya.

2.2.5 Definisi Pemantauan

Menurut *Oxfam* Pemantauan adalah mekanisme yang sudah menyatu untuk memeriksa bahwa semua berjalan untuk direncanakan dan memberi kesempatan agar penyesuaian dapat dilakukan secara metodologis. Sedangkan Menurut *WHO*, Pemantauan adalah suatu proses pengumpulan dan menganalisis informasi dari penerapan suatu program termasuk mengecek secara reguler untuk melihat apakah kegiatan / program itu berjalan sesuai rencana sehingga masalah yang dilihat / ditemui dapat diatasi.

2.2.6 Definisi Perilaku

Perilaku yaitu suatu respon seseorang yang dikarenakan adanya suatu stimulus / rangsangan dari luar (Notoatmodjo, 2012). Perilaku dibedakan menjadi dua yaitu perilaku tertutup (*covert behavior*) dan perilaku terbuka (*overt behavior*). Perilaku tertutup merupakan respon seseorang yang belum dapat diamati secara jelas oleh orang lain. Sedangkan perilaku terbuka merupakan respon dari seseorang dalam bentuk tindakan yang nyata sehingga dapat diamati lebih jelas dan mudah (Fitriani, 2011). Bentuk perilaku dilihat dari sudut

pandang respon terhadap stimulus, maka perilaku dapat dibedakan menjadi dua yaitu:

1. Perilaku Tertutup, Perilaku tertutup adalah respon seseorang terhadap stimulus dalam bentuk terselubung atau tertutup. Respon atau reaksi terhadap stimulus ini masih terbatas pada perhatian, persepsi, pengetahuan / kesadaran, dan sikap yang terjadi belum bisa diamati secara jelas oleh orang lain.
2. Perilaku Terbuka, Perilaku terbuka adalah respon seseorang terhadap stimulus dalam bentuk tindakan nyata atau terbuka. Respon terhadap terhadap stimulus tersebut sudah jelas dalam bentuk tindakan atau praktek (*practice*).

2.2.7 Sejarah SMA Negeri 1 Bukit Kemuning

SMAN 1 Bukit Kemuning adalah Sekolah Menengah Atas (SMA) Negeri yang berlokasi di Propinsi Lampung Kabupaten Kab.Lampung Utara dengan alamat Jl. Baturaja LK – IX, Kec. Bukit Kemuning, Kab. Lampung Utara, Prov. Lampung.SMA Negeri 1 Bukit Kemuning pertama kali didirikan berdasarkan Surat Keputusan Menteri Dikbud Nomor. 0292/0/1978 tanggal 2 September 1978 berupa pembukaan sekolah baru dengan nama SMA Negeri Bukit Kemuning, berlaku mulai tanggal 1 April 1978. Pada tahun 2010, SMA Negeri 1 Bukit kemuning mendapatkan akreditasi “A” berdasarkan SK Akreditasi BAN No.Ma 008430 tanggal 22 November 2010.

2.2.8 Definisi Website

(Raymod M. Jr. & George P. S., 2014) Kategori perangkat lunak yang berpusat pada jaringan komputer ini menyediakan sederetan luas aplikasi – aplikasi. Sederhananya, aplikasi – aplikasi *Web* sedikit lebih dari sekedar sekumpulan *file hypertext* yang saling terhubung untuk menunjukkan informasi – informasi tertentu dengan menggunakan grafis – grafis yang sifatnya terbatas dan menampilkan informasi – informasi tertentu dengan teks – teks. Namun, seiring

munculnya *Web 2.0*, aplikasi – aplikasi *web* saat ini sedang berevolusi menjadi lingkungan komputasi yang canggih yang tidak hanya menyajikan fitur – fitur mandiri, fungsi – fungsi komputasi, dan isi untuk pengguna akhir, namun juga terintegrasi dengan sistem basis data yang dimiliki oleh perusahaan dan juga terintegrasi dengan aplikasi – aplikasi bisnis lainnya. Berikut ini merupakan tujuan dari adanya *website* :

1. Sarana presentasi Profil Perusahaan yang bisa diakses di seluruh Indonesia dan Dunia.
2. Sarana promosi Produk dan Jasa.
3. Media Komunikasi antar individu dan saling bertukar informasi misalnya jejaring sosial.
4. Media ekspresi pribadi maupun kelompok.
5. Menyediakan sarana penyebar Ilmu Pengetahuan.
6. Tempat untuk belajar baik dengan interaktif langsung maupun Forum.
7. Sarana berbagi file baik berupa program maupun data.
8. Sebagai alat transaksi *Online* sehingga memudahkan melakukan transaksi dimana saja dan kapan saja.

2.2.9 Bahasa Pemrograman dan Perangkat Lunak Pendukung

Bahasa pemrograman yang digunakan dalam penulisan skripsi ini adalah *PHP* sedangkan perangkat lunak pendukung yang digunakan adalah *MySQL*, *XAMPP*, *HTML5*, *CSS*, *Google Maps API*, dan *Sublime Text Editor*.

a. PHP (*Hypertext Preprocessor*)

(RohiAbdulloh, 2015) PHP singkatan dari *Hypertext Preprocessor* yang merupakan *server side programming*, yaitu Bahasa pemrograman yang diproses disisi server. Fungsi utama PHP dalam membangun *website* adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke database, diedit, dihapus, dan ditampilkan pada *website* yang diatur oleh PHP.

Pengembangan demi pengembangan terus berlanjut, ratusan fungsi ditambahkan sebagai fitur dari bahasa PHP, dan diawal tahun 1999, *netcraft* mencatat, ditemukan 1.000.000 situs didunia telah menggunakan PHP. Ini membuktikan bahwa PHP merupakan bahasa yang paling populer digunakan oleh dunia *web development*. Hal ini mengagetkan para developernya termasuk Rasmus sendiri, dan tentunya sangat diluar dugaan sang pembuatnya. Kemudian Zeev Suraski dan Andi Gutsman selaku *core developer* (program merinti) mencoba untuk menulis ulang PHP Parser, dan di integrasikan dengan menggunakan *Zend Scripting Engine*, dan mengubah jalan alur operasi PHP. Dan semua fitur baru tersebut dirilis dalam PHP4. 13 Juli 2004, evolusi PHP, PHP telah mengalami banyak sekali perbaikan disegala sisi, dan wajar jika *netcraft* mengumumkan PHP sebagai bahasa *web* populer di dunia, karena tercatat 19 juta domain telah menggunakan PHP sebagai *server side scripting* nya.

b. *JQuery*

(Loka Dwiartara, 2013) *jQuery* adalah pustaka kecil dari *JavaScript* bersumber terbuka yang menekan kan pada interaksi antara *Javascript* dan *HTML*. Pustaka ini dirilis pada Januari 2006 di Bar Camp NYC oleh John Resig dan berlisensi ganda dibawah Lisensi MIT dan GPL. *jQuery* sangat ringan walaupun hanya sedikit tulisan namun *jQuery* mampu melakukan hal lebih menakjubkan. *jQuery* mampu membuat tampilan suatu *web* menjadi lebih menarik. *JQuery* adalah sebuah perpustakaan *Java script* berbeban ringan "sedikit menulis, banyak kerja". Perpustakaan *jQuery* memuat fitur – fitur berikut :

- a) Seleksi elemen HTML.
- b) Manipulasi elemen HTML.
- c) Manipulasi CSS.
- d) Fungsi – fungsi *event* HTML.

c. *MySQL*.

(Loka Dwiartara) *MySQL* bersifat RDBMS (*Relational Database Management System*) yang memungkinkan seorang admin dapat menyimpan banyak informasi ke tabel – tabel, dimana tabel – tabel tersebut saling berkaitan satu sama lain. Keuntungan RDBMS sendiri adalah kita dapat memecah database kedalam tabel – tabel yang berbeda, setiap tabel memiliki informasi yang berkaitan dengan tabel yang lainnya. *MySQL* adalah *multiuser database* yang menggunakan bahasa *Structured Query Language (SQL)*. *MySQL* dalam operasi *client – server* melibatkan *server daemon MySQL* disisi *server* dan berbagai macam program serta *library* yang berjalan di sisi *client*. *MySQL* mampu menangani data yang cukup besar. Perusahaan yang mengembangkan *MySQL* yaitu TeX, mengaku mampu menyimpan data lebih dari 40 *database*, 10.000 tabel dan sekitar 7 juta baris, totalnya kurang lebih 100 *Giga Byte* data. (Di akses pada www.mysql.com).

d. *XAMPP*

(Andi, 2009) *XAMPP* adalah salah satu paket *Software Web Server* yang terdiri dari *Apache*, *MySQL*, *PHP* dan *PHPMyAdmin*. Mengapa menggunakan *XAMPP* ? Karena *XAMPP* sangat mudah penggunaanya, terutama jika Anda seorang pemula. Proses instalasi *XAMPP* sangat mudah, karena tidak perlu melakukan konfigurasi *Apache*, *PHP* dan *MySQL* secara manual, *XAMPP* melakukan instalasi dan konfigurasi secara otomatis.

e. HTML (*Hyper Text Markup Language*)

(Rosa A. S, M. Shalahuddin, 2014) HTML singkatan dari *Hyper Text Markup Language*, yaitu skrip yang mengatur berupa *tag – tag* untuk membuat dan mengatur struktur *website*.

Beberapa tugas utama HTML dalam membangun *website* diantaranya sebagai berikut :

- a) Menentukan *Layout Website*.
- b) Memformat *Text* dasar seperti pengaturan paragraf, dan format font.
- c) Membuat *List*.
- d) Membuat tabel.
- e) Menyisipkan gambar, video, dan audio.
- f) Membuat *Link*.
- g) Membuat formulir.

2.2.10 Definisi UML (*Unified Modelling Language*)

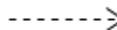
(Booch, 2014) UML merupakan suatu bahasa. Suatu bahasa terdiri dari kata – kata dan memiliki aturan untuk menggabungkan kata – kata tersebut, sehingga tercipta komunikasi. Sebuah permodelan bahasa adalah suatu bahasa dimana kata – kata dan aturannya berfokus pada penggambaran sistem secara konseptual dan fisik. Sebuah permodelan bahasa seperti UML telah menjadi bahasa standar untuk merencanakan suatu aplikasi. UML terdiri dari beberapa Tipe *diagram*, antara lain :

a. *Use Case Diagram*

Use case atau diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sistem informasi dan oleh siapa saja yang berhak menggunakan fungsi – fungsi itu. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel

mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

Tabel 2.1 Simbol *Use Case Diagram*

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.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.

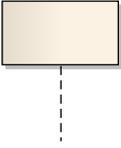
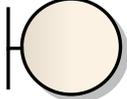
8		<i>Use Case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan – aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen – elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Sumber : (Rosa Salahudin, 2012)

b. *Sequence Diagram*

Sequence Diagram menggambarkan kelakuan objek pada *Use Case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan *Sequence Diagram* maka harus diketahui objek – objek yang terlibat dalam sebuah *Use Case* beserta metode – metode yang dimiliki kelas yang diinstansi menjadi objek itu. Banyaknya *Sequence Diagram* yang harus digambar adalah sebanyak pendefinisian *Use Case* yang memiliki proses sendiri atau yang penting semua *Use Case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *Sequence Diagram* yang harus dibuat juga semakin banyak. Berikut adalah simbol – simbol yang ada pada *Sequence Diagram*.

Tabel 2.2 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Lifeline</i>	Objek entity, antarmuka yang saling berinteraksi.
2.		<i>Actor</i>	Digunakan untuk menggambar kan <i>user</i> / pengguna.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memiliki informasi – informasi tentang aktifitas yang terjadi.
4.		<i>Boundary</i>	Digunakan untuk menggambar kan sebuah form.
5.		<i>Control Class</i>	Digunakan untuk menghubungkan boundary dan tabel.
6.		<i>Entity Class</i>	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.

Sumber : (A. S & Shalahuddin, 2012)

c. *Activity Diagram*

Diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *Diagram* aktivitas juga banyak digunakan untuk mendefinisikan hal – hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.

2. Urutan pengelompokkan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak. Sangat bermanfaat apabila kita membuat diagram ini terlebih dahulu dalam memodelkan sebuah proses untuk membantu memahami proses secara keseluruhan.

Tabel 2.3 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>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

Sumber : (A. S & Shalahuddin, 2012)

d. *Class Diagram*

Class Diagram adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki 3 bagian utama yaitu *attribute*, *operation*, dan *name*. Kelas – kelas yang ada pada struktur sistem harus dapat melakukan fungsi – fungsi

sesuai dengan kebutuhan sistem. Berikut ini merupakan simbol dalam *class diagram*.

Tabel 2.4 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 mempegaruhi elemen yang bergantung pada elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Sumber : (A. S & Shalahuddin, 2012)

2.2.11 Pengertian *Black Box Testing*

Menurut Shalahuddin dan Rosa (2011), *Black Box Testing* adalah menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi – fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang di buat untuk melakukan pengujian *black box testing* harus dibuat dengan kasus benar dan kasus salah.