

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam pembuatan sistem informasi penyewaan mobil di Ajib Rental Sukoharjo ada beberapa penelitian yang terkait sebagai dasar acuan dalam penelitian ini.

Penelitian yang dilakukan oleh Fitria dan Nisa (2017), tentang Sistem Informasi penyewaan bus *online* berbasis *web* pada dinas perhubungan Kabupaten Barito Kuala. Pada penelitian ini sistem yang akan dibangun menggunakan *PHP*, *MySQL* dan *XAMPP* sebagai *server*-nya. Didalam aplikasi itu terdapat menu halaman utama, *login admin*, *form* tarif sewa, jadwal sewa, daftar pemesan. Penelitian ini menggunakan metode wawancara, data sekunder. Aplikasi Penyewaan dibuat untuk mempermudah *customer* dalam melakukan penyewaan bus dan penjadwalan agar semua data-data yang diperoleh dapat tersusun rapi dalam *database*.

Penelitian yang kedua dilakukan oleh Kristiawan dan Wardati (2015). Tentang pembuatan *website* pemesanan mobil pada rental mobil Akur Pacitan. Pada penelitian ini dilakukan pembangunan menggunakan *PHP* dan *MySQL*. Dalam aplikasi ini terdapat menu halaman login admin, daftar mobil, cara pemesanan, produk, daftar pemesanan, tentang kami. Aplikasi pemesanan mobil berbasis *website* supaya dapat memfasilitasi pelanggan rental mobil untuk memesan mobil di rental Akur di Pacitan secara *online*.

Penelitian ke tiga yang dilakukan Septavia dan Gunadhi (2015), menciptakan sistem informasi penyewaan mobil berbasis *web* di Jasa Kurnia Tour and Travel. Pada penelitian ini dilakukan pembangunan menggunakan Metodologi yang digunakan adalah metodologi Waterfall yang terdiri dari tahapan Analisa Kebutuhan, Desain Sistem, *Coding*, Pengujian Program, Pemeliharaan. Sedangkan bahasa pemrograman yang digunakan yaitu bahasa pemogramaman *PHP* dan untuk databasenya menggunakan *MySQL*. Dari hasil penyusunan ini, rancang bangun Aplikasi Penyewaan Mobil ini bisa mengakomodasi kebutuhan

dalam kegiatan transaksi sewa, pemesanan, pembayaran sehingga perusahaan dapat memperoleh informasi yang lebih akurat dari setiap aktivitas yang dikerjakan.

Kelemahan dari ketiga adalah belum adanya sistem transaksi dalam pembayaran pada saat penyewaan kendaraan, dan pelanggan harus melakukan *login* akun terlebih dahulu sebelum pemesanan.

2.2 Kerangka Pemikiran

Kerangka pemikiran yang dijalankan dalam penelitian ini adalah sebagai berikut :

1. Latar Belakang Masalah

Penyewaan mobil di Ajib Rental masih secara *offline* sehingga penyewa membutuhkan waktu yang cukup lama untuk menyewa mobil karena harus datang langsung atau menelpon pihak pengelola dan tidak bisa mengetahui stok ketersediaan mobil dengan cepat.

2. Rumusan Masalah

Bagaimana membuat Sistem Informasi Penyewaan mobil di Ajib Rental.

3. Penguasaan Dasar *PHP*, dan *MySQL*

Tahap untuk mempelajari dasar-dasar *PHP* dan *MySql* agar lebih menguasai program-program yang akan digunakan untuk membangun sistem.

4. Pengumpulan Data Tertulis dan Tidak Tertulis

Tahap pengumpulan data pada penelitian ini melalui observasi, wawancara dan studi literatur. Pengumpulan data bertujuan untuk mengetahui permasalahan dan kebutuhan informasi mengenai penyewaan mobil di Ajib Rental.

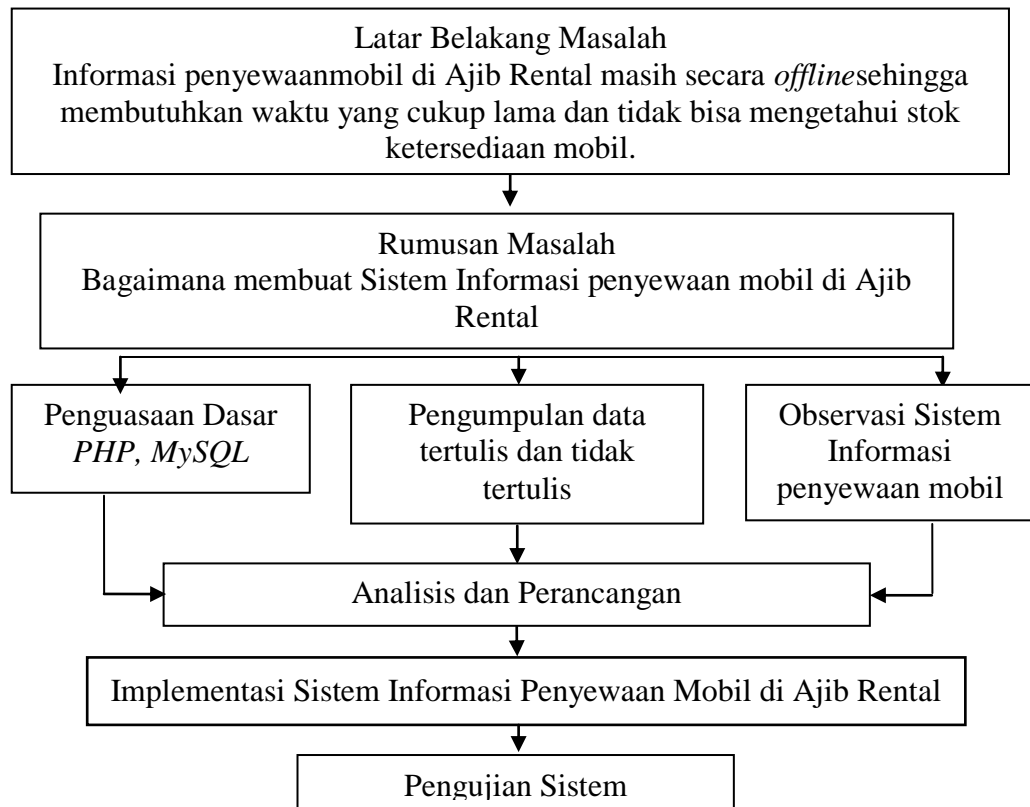
5. Observasi Sistem Informasi penyewaan

Merupakan tahap pengamatan contoh sistem informasi yang telah ada, jurnal, buku, maupun karya ilmiah untuk kajian yang dapat dijadikan referensi untuk pembangunan sistem.

6. Analisis dan Perancangan aplikasi terstruktur.

7. Implementasi Sistem Informasi Penyewaan mobil di Ajib Rental Sukoharjo.
8. Implementasi sistem adalah langkah-langkah atau prosedur-prosedur yang dilakukan dalam menyelesaikan desain sistem yang telah disetujui, untuk menginstal, menguji dan memulai sistem baru atau sistem yang diperbaiki. Lingkungan implementasi *website* ini meliputi kebutuhan perangkat lunak, perangkat keras, *form* program yang sesuai, *query* yang digunakan, pemrograman, pengujian program dan pengujian Sistem Informasi yang telah dirancang sesuai dengan kebutuhan dalam pembuatan sistem informasi penyewaan mobil ini.
9. Pengujian Sistem
Pengujian sistem adalah prosedur yang dilakukan untuk menyelesaikan rancangan sistem yang telah disetujui, menguji sistem, menginstal serta memulai penggunaan sistem baru atau sistem yang telah diperbaiki. Dalam pengujian sistem ini menggunakan metode *black box testing*..

Berdasarkan penjabaran dari kerangka pemikiran penelitian tentang sistem informasi Penyewaan mobil di Ajib Rental Sukoharjo, maka dapat digambarkan diagram alur seperti yang ditunjukkan pada Gambar 2.1.



Gambar 2.1. Kerangka Pemikiran

2.3 Landasan Teori

Penjelasan dari beberapa istilah yang berhubungan dengan pembuatan Sistem Informasi Penyewaan mobil di Ajib Rental Sukoharjo.

2.3.1 Pengertian Sistem

Menurut Elisabet (2017), Sistem adalah kumpulan orang yang saling bekerja sama dengan ketentuan-ketentuan aturan yang sistematis dan terstruktur untuk membentuk satu kesatuan yang melaksanakan suatu fungsi untuk mencapai tujuan. Sistem memiliki beberapa karakteristik atau sifat yang terdiri dari komponen sistem, batasan sistem, masukan sistem, keluaran sistem, pengolahan sistem dan sasaran sistem.

2.3.2 Pengertian Informasi

Menurut Elisabet (2017), Informasi merupakan data yang diolah menjadi lebih berguna dan berarti bagi penerimanya, serta untuk mengurasi ketidakpastian dalam proses pengambilan keputusan mengenai suatu keadaan.

2.3.3 Pengertian Sistem Informasi

Menurut Abdul (2003), Sistem Informasi adalah sejumlah komponen (manusia, komputer, teknologi, informasi, dan prosedur kerja), ada suatu sasaran atau tujuan. Setelah menguraikan definisi dari sistem informasi maka secara umum sistem informasi dapat diartikan sebagai kumpulan dari berbagai sub siste, baik lainnya yang berjalan secara harmonis untuk mencapai tujuan yang sama yaitu mengumpulkan dan mengelola data menjadi suatu informasi yang bermanfaat dan memiliki sebuah nilai

Menurut Mamed (2015), dalam arti yang luas sistem informasi dapat dipahami sebagai sekumpulan sub sistem yang saling berhubungan, berkumpul bersamasama dan membentuk satu kesatuan, saling berinteraksi dan bekerja sama antara bagian satu dengan yang lainnya dengan cara-cara tertentu untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi sebagai dasar bagi pengambilan keputusan yang berguna dan mempunyai nilai nyata yang dapat dirasakan akibatnya baik pada saat itu juga maupun di masa mendatang, mendukung kegiatan operasional, manajerial, dan strategis organisasi, dengan memanfaatkan berbagai sumber daya yang ada dan tersedia bagi fungsi tersebut guna mencapai tujuan.

Berdasarkan komponen fisik penyusunannya, sistem informasi terdiri atas komponen berikut :

1. Perangkat keras (*Hardware*) Perangkat keras dalam sistem informasi meliputi perangkatperangkat yang digunakan oleh sistem komputer untuk masukan dan keluaran (*input dan output*), memori, modem, pengolah (*processor*), dan periferal lainnya.

2. Perangkat lunak (*Software*) Perangkat lunak dalam sistem informasi adalah berupa program-program komputer yang meliputi sistem operasi (*operating system/OS*), bahasa pemrograman (*programming language*), dan program-program aplikasi.
3. Berkas basis data (*File*) Berkas merupakan sekumpulan data dalam basis data yang disimpan dengan cara-cara tertentu sehingga dapat digunakan kembali dengan mudah dan cepat.
4. Prosedur (*Procedure*) Prosedur meliputi prosedur pengoperasian untuk sistem informasi, manual, dan dokumen-dokumen yang memuat aturan-aturan yang berhubungan dengan sistem informasi dan lainnya.
5. Manusia (*Brainware*) Manusia yang terlibat dalam suatu sistem informasi meliputi operator, programmer, sistem analis, manajer sistem informasi, manajer pada tingkat operasional, manajer pada tingkat manajerial, manajer pada tingkat strategis, teknisi, administrator basis data (*Database Administrator/DBA*), (jaringan komunikasi), dan *database* (basis data) yang mengumpulkan, mengubah dan menyebarkan informasi di dalam suatu bentuk organisasi.

2.3.4 Pengertian Rental Mobil

Menurut Intan dan Erwin (2015), Rental mobil merupakan sebuah usaha yang sangat pesat perkembangannya. Dengan roda kehidupan yang sangat cepat membuat orang berlomba-lomba dengan cepat pula untuk mencapai tempat tujuan yang diinginkan dengan mudah, nyaman dan murah serta dengan tidak menurunkan gengsi, derajat dan martabat orang tersebut.

Rental mobil adalah penyedia layanan penyewaan mobil dengan cara sewa harian ataupun kontrak dengan menggunakan *driver* ataupun lepas kunci, pemanfaatan rental mobil ini dapat dikembangkan sebagai terobosan bagi masyarakat atau perusahaan yang tidak memiliki alat transportasi yang akan digunakan untuk oprasiaonal. Sesuai dengan namanya, rental mobil harian maka pemakaian yang dihitung adalah dalam jangka harian atau perhari. Pada jenis layanan ini terdapat dua kategori yaitu rental mobil dengan sopir atau tanpa sopir

(biasa disebut rental mobil lepas kunci).Paket rental mobil perjam yaitu bagi yang ingin menggunakan jasa sewa mobil singkat layanan ini yang pas untuk memenuhi kebutuhan. Paket sewa mobil perjam biasanya dibatasi mulai dari paket sewa mobil 6 jam dan 12 jam.

Salah satu alat transportasi yang memiliki fungsi yang baik dan dengan kapasitas angkut yang banyak, serta mudah dan murah untuk dibawa dan disewa adalah mobil.Dalam perkembangannya, usaha penyewaan mobil sudah menjadi usaha atau bisnis yang sangat menguntungkan. Karena pada saat sekarang ini masyarakat lebih senang menggunakan mobil untuk bepergian jauh maupun dalam jarak yang sedang untuk ditempuh. Sehingga rasa senang dalam perjalanan akan lebih mudah didapatkan jika dibandingkan dengan menggunakan alat transportasi lainnya.

Keberhasilan suatu usaha penyewaan mobil dapat diukur berdasarkan tinggi rendahnya hasil atau laba yang diperoleh perusahaan tersebut. Semakin cepat, tepat dan akurat suatu sistem dalam perusahaan maka akan berdampak pada pelayanan yang memuaskan serta menghasilkan informasi yang akurat untuk manajemen.

2.3.5 Website

Menurut Mayadewi dan Paramita, Juanda (2016), *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).Awalnya suatu *web* hanya dapat menyajikan informasi, saat ini suatu *web* telah dapat berinteraksi dengan pengguna melalui pengisian *form* validasi input atau tansaksi *online*.

2.3.6 HTML (*Hypertext Markup Language*)

Menurut Wardati dan Kristiawan (2015), *Hypertext Markup Language* adalah merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan di halaman *web*. Oleh karena itu agar dapat membuat program aplikasi diatas halaman *web*, anda terlebih dahulu harus mengenal dan menguasai HTML .

2.3.7 PHP (*Perl Hypertext Preprocessor*)

Menurut Wirawan (2016), PHP bahasa pemrograman yang berfungsi untuk membuat *website dinamis* maupun aplikasi *web*. Berbeda dengan *HTML* yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan database, *file* dan folder, contohnya Blog, Toko *Online*, *CMS*, Forum, dan *Website Social Networking*. *PHP* adalah bahasa scripting, bukan bahasa *tag-based* seperti *HTML*. *PHP* termasuk bahasa *cross-platform*, ini artinya PHP bisa berjalan di sistem operasi yang berbeda-beda (*Windows, Linux, ataupun MAC*) “Untuk dapat berjalan, PHP membutuhkan *web server*, yang bertugas untuk memproses file php dan mengirimkan hasil pemrosesan yang akan ditampilkan di *browser client*. Oleh karena itu, PHP termasuk server-side scripting (script yang diproses di server). Web server sendiri adalah software yang diinstal di komputer lokal ataupun komputer lain yang berada di jaringan intranet/internet yang berfungsi untuk melayani permintaan-permintaan web dari *client*. *Web server* yang paling digunakan saat ini untuk PHP adalah “*Apache*”. Untuk media penyimpanan datanya (*database server*), PHP biasa menggunakan *MySQL*”. “Untuk menginstall dan mengkonfigurasi ketiga software tersebut (*Apache, PHP, MySQL*) agar dapat berjalan dan selalu terhubung, memang cukup sulit. Maka dari itu dibuatlah paket *software LAMP, XAMPP, MAMP, WAMP* yang tinggal kita install dalam satu kali instalasi. Dalam satu kali instalasi, sudah mencakup ketiga *software* tersebut dan sudah dikonfigurasi untuk keperluan lingkungan pengembangan aplikasi web”.

2.3.8 Database

Menurut Rosa dan Shalahuddin (2015), Basis data (*Database*) merupakan salah satu bagian dalam rekayasa perangkat lunak yang terkomputerisasi dan bertujuan utama memelihara data yang sudah diolah atau media penyimpanan informasi agar dapat diakses dengan mudah dan cepat.

2.3.9 MySQL

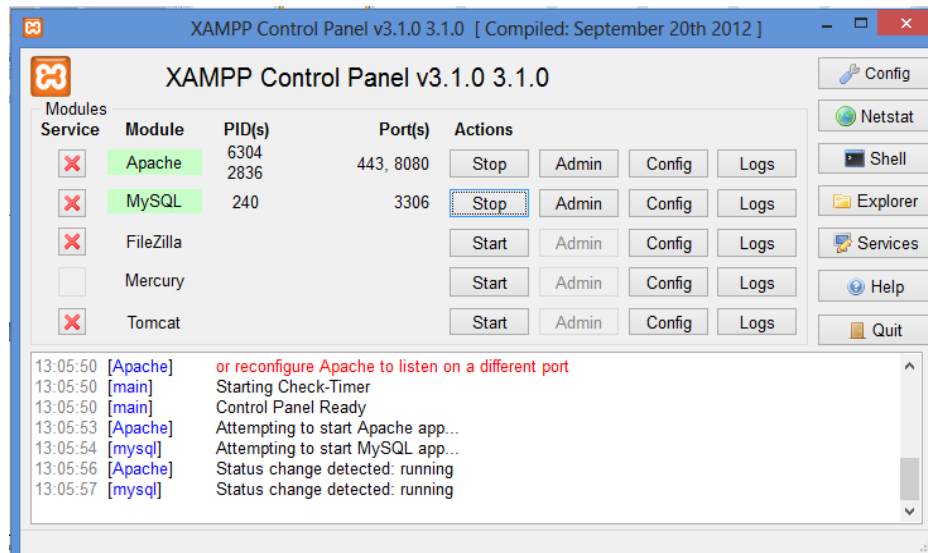
Menurut Wardati dan Kristiawan (2015), *MySQL* adalah *multiuser* database yang menggunakan bahasa *Structured Query Language (SQL)*. *MySQL* dalam operasi *client server* melibatkan *server daemon MySQL* disisi server dan berbagai macam program serta library yang berjalan disisi client. *MySQL* mampu menangani data yang cukup besar. Perusahaan yang mengembangkan *MySQL* yaitu TEX, mengaku mampu menyimpan data lebih dari 40 database, 10.000 tabel, dan sekitar 7.000.000 baris totalnya kurang lebih 100 *Gigabyte* data.

Pada bulan Mei 1996, *MySQL* versi 1.0 berhasil dirilis namun penggunaannya terbatas 4 orang saja. Namun di bulan Oktober di tahun yang sama versi 3.11.0 dilepaskan ke publik tapi belum bersifat *open source*. Bulan Juni 2000, *MySQL AB* mengumumkan bahwa sejak versi 3.23.19, *MySQL* merupakan *software database* yang bebas berlisensi GPL atau *General Public License* yang *open source*. Mulanya *MySQL* hanya berjalan di sistem operasi *linux* namun pada saat *MySQL* versi 3.22 tahun 1998-1999 sudah tersedia diberbagai *platform* termasuk *windows*. Ini terjadi karena *MySQL* menjadi semakin populer dan dilirik banyak orang karena kestabilan dan kecepatan yang meningkat.

2.3.10 XAMPP

Menurut Bunafit (2011) bahwa *XAMPP* merupakan merupakan paket *PHP* berbasis *open source* yang dikembangkan oleh sebuah komunitas *Open Source*. Dengan menggunakan *XAMPP* kita tidak perlu lagi melakukan penginstalan program yang lain karena semua kebutuhan telah disediakan oleh *XAMPP*. Beberapa pakat yang telah disediakan adalah *Apache*, *MySql*, *Php*, *Filezila*, dan *Phpmyadmin* .

Menurut Riyanto (2011), Xampp merupakan paket *PHP* dan *MySQL* berbasis *open source*, yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis *PHP*. *Xampp* mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket. *XAMPP* ditunjukkan pada gambar 2.2 dibawah ini:



Gambar 2.2 Contoh *XAMPP*

2.3.11 Perancangan Sistem

Menurut Hanif dan Robert (2015), Perancangan sistem mulanya diawali dengan menentukan segala keperluan yang akan memenuhi apa yang dibutuhkan oleh sistem, siapa yang mengambil langkah dan bagaimana cara menyesuaikan. Pada dasarnya perancangan sistem bergerak dari input menuju ke *output* sistem, yang terdiri dari *reports* dan *file* untuk memenuhi kebutuhan organisasi. Perancangan sistem merupakan sebuah penentuan proses data yang diperlukan oleh sistem baru, dan tahap-tahap dalam merancang sistem, meliputi:

1. Menyiapkan perancangan sistem secara rinci
Analisis berkerjasama dengan pemakai dan mendokumentasikan rancangan sistem baru menggunakan peralatan tertentu.
2. Mengidentifikasi alternatif konfigurasi sistem
3. Analisis harus mengidentifikasi konfigurasi peralatan komputer yang memberi hasil sesuai dengan yang diperlukan untuk menyelesaikan proses.

4. Mengevaluasi alternatif konfigurasi sistem
Analisis berkerjasama dengan manager untuk mengevaluasi alternatif.
5. Memilih konfigurasi terbaik.
6. Menyiapkan Usulan Implementasi.
7. Menyiapkan usulan penerapan yang memberi ringkasan tugas-tugas penerapan yang harus dilakukan dari dokumentasi perancangan.
8. Menyetujui dan menolak penerapan sistem.

2.3.12 UML (*Unified Modeling Language*)

Menurut Rosa dan Shalahudin (2016), *Unified Modeling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. *UML* merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

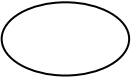
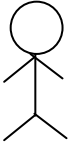

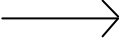

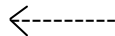
UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan *UML* tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya *UML* paling banyak digunakan pada metodologi berorientasi objek.

1. *Use case Diagram*

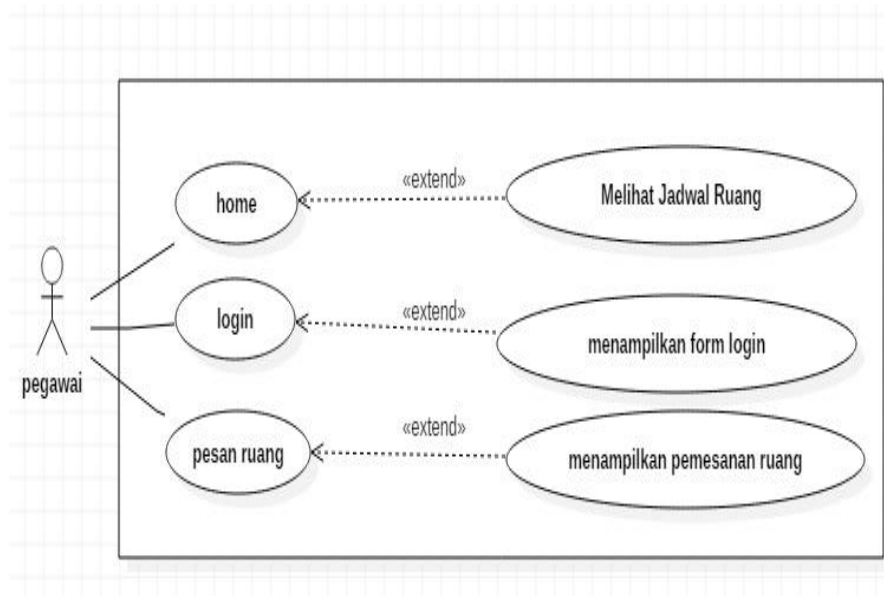
Menurut Kusnita (2016) *Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use-case diagram* dapat digunakan selama proses analisis untuk menangkap *requirements system* dan untuk memahami bagaimana sistem seharusnya bekerja. “selama tahap desain, *use-case diagram* berperan untuk menetapkan perilaku sistem saat diimplementasikan. *Use case diagram* juga memiliki komponen seperti. Aktor sistem lain yang berinteraksi dengan sistem yang akan dibuat, usecase sistem sebagai unit bertukar pesan antar unit maupun aktor, dan relasi sistem baik aktor

maupun antara *usecase* dengan *usecase* dan aktor. Simbol-simbol yang digunakan dalam *use case* diagram yaitu diunjukkan pada Tabel 2.1:

Tabel 2.1 Simbol *Use Case Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Use Case</i>	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja.
2.		<i>Actor</i>	<i>Actor</i> merupakan orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi <i>actor</i> , harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target system
3.		<i>Association</i>	<i>Association</i> antara <i>actor</i> dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
4.		<i>Generalization</i>	<i>Generalization</i> merupakan <i>association</i> antara <i>actor</i> dan <i>use case</i> yang menggunakan panah terbuka untuk mengidentifikasi bila aktor berinteraksi secara pasif dengan sistem.
5.		<i>Include</i>	<i>Include</i> merupakan pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
6.		<i>Extend</i>	<i>Extend</i> merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

Contoh *Usecase* diagram



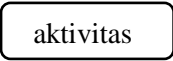

Gambar 2.3 Contoh *Usecase Diagram*

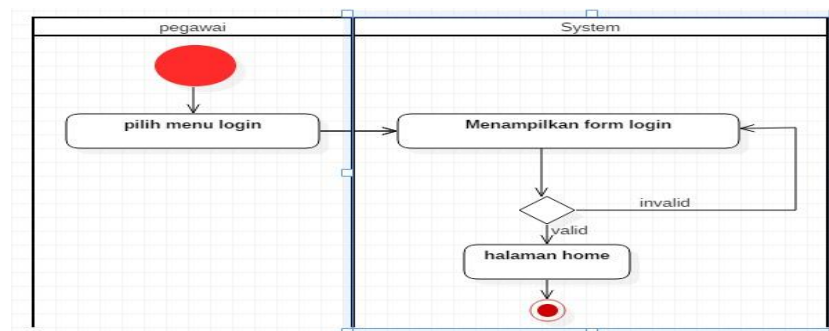
Pada gambar 2.3 menjelaskan bahwa terdapat 3 menu pilihan yaitu *home*, *login* dan *pesan ruang*. Jika pengguna memilih menu *home*, maka akan menampilkan daftar *list* ruangan jadwal ruangan, *login* menampilkan *form login* dan *pesan ruang* menampilkan pemesanan ruang.

2. *Activity Diagram*

Menurut Rosa&Salahudin (2016), *Activity diagram* menggambarkan *workflow* atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol *Activity Diagram* ditunjukkan pada Tabel 2.2 dibawah ini:

Tabel 2.2 Simbol-simbol *Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		Status awal	Status awal aktivitas sistem.
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		<i>Decision</i>	Asosiasi percabang di mana jika adapili han aktivitas lebih dari satu.
4		<i>Join</i>	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu.
5		Status akhir	Status akhir yang dilakukan sistem.
6		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Contoh dari *Activity Diagram*Gambar 2.4 Contoh *Actifity Diagram*

Pada Gambar 2.4 menjelaskan pada saat pegawai memilih menu *login*, maka sistem akan menampilkan ke *form login*.



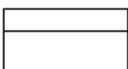

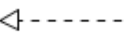
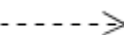

3. *Class Diagram*

Menurut Rosa dan Shalahudin (2016), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akandibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan *method* atau operasi. Berikut penjelasan atribut dan *method* :

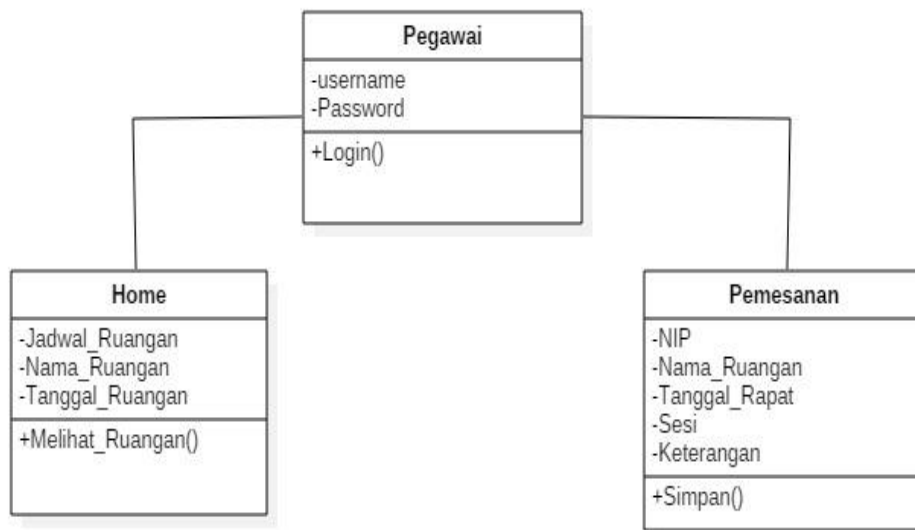
1. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Berikut adalah simbol-simbol yang ada pada diagram kelas ditunjukkan pada Tabel 2.3:

Tabel 2.3 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 padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Contoh dari *Class Diagram*



Gambar 2.5 Contoh *Class Diagram*



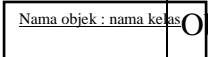

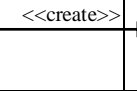
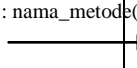
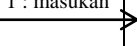
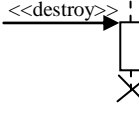
Class diagram pada *website* pemesanan ruang rapat di Diskominfo Statistik dan Persandian Surakarta petugas memiliki atribut *username* dan *password* yang dapat digunakan untuk *login website*. Dan memiliki 2 Class yaitu home dan pemesanan. Didalam home terdapat atribut jadwal ruangan, nama ruangan, tanggal ruangan yang dapat melihat ruangan. Dan dalam class pemesanan terdapat atribut nip, nama ruang, tanggal rapat, sesi, dan keterangan agar bisa menyimpan inputan.

4. *Sequence Diagram*

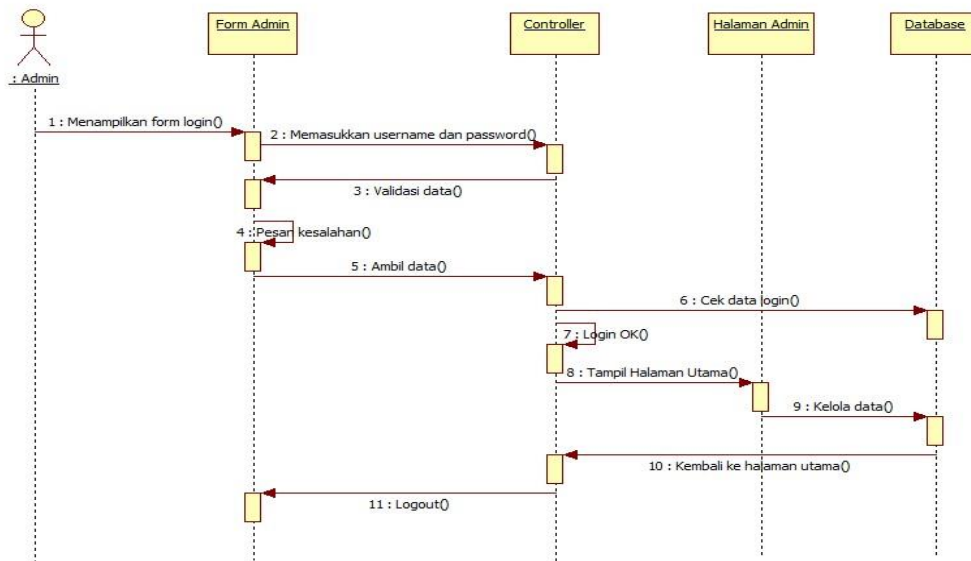
Menurut Rosa & Shalahuddin (2016). *Sequence* diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Dalam *UML*, objek, pada diagram *sequence* digambarkan dengan segi empat, yang beris nama dari objek yang digarisbawahi. Terdapat tiga cara untuk menamai objek yaitu, nama objek, nama objek dan *class* sertanama *class*. Perlu diingat bahwa di dalam diagram ini, kelas-kelas dan *actor-actor* diletakkan di bagian atas diagram dengan urutan dari kiri ke kanan dan garis *lifeline* yang diletakkan secara vertikal

terhadap kelas dan aktor. Berikut ini adalah simbol *sequence* diagram dapat dilihat pada Tabel 2.4 :

Tabel 2.4. Tabel *Sequence Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frasa nama aktor.
2		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3		Objek	Menyatakan objek yang berinteraksi pesan.
4		Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi.
5		Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arahnya mengarah pada objek yang dibuat
6		Pesan tipe call	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.
7		Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arahnya mengarah pada objek yang dikirim.
8		Pesan tipe destroy	Menyatakan suatu objek mengakhiri hidup objek lain, arahnya mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.

Contoh Gambar *Sequence Diagram*



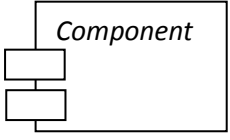

Gambar. 2.6 Contoh *Sequence Diagram*

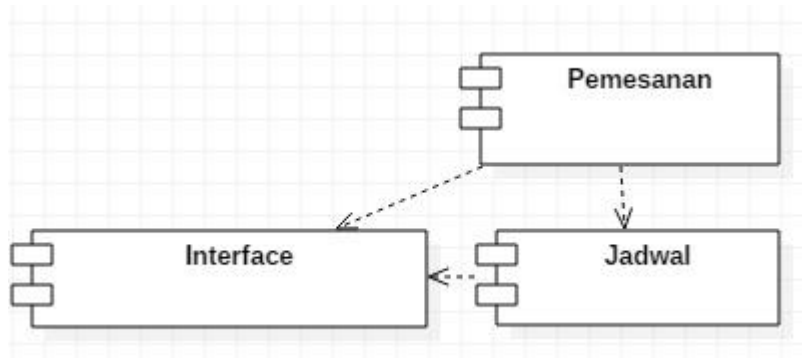
Pada Gambar 2.6 menjelaskan tentang *Sequence Diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

5. *Component Diagram*

Menurut *Component diagram* menggambarkan struktur fisik dari kode, pemetaan pandangan logis dari kelas proyek untuk kode aktual di mana logika ini dilaksanakan. Simbol-simbol yang digunakan dalam *component diagram* antara lain ditunjukkan pada Tabel 2.4 *Component Diagram*

Tabel 2.5 Simbol *Component Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Component</i>	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.		<i>Dependency</i>	Sebuah <i>Dependency</i> digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan ke pada komponen tempat sebuah komponen itu bergantung.

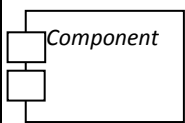
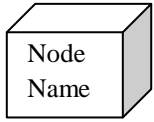

Contoh *Component Diagram*Gambar 2.7 contoh *Component Diagram*

Pada Gambar 2.7 contoh *Component Diagram website* pemesanan ruang rapat, sedangkan interface atau tampilan dari website ini mencakup jadwal ruangan dan pemesanan ruang rapat pada sistem tersebut.

6. Deployment Diagram

Menurut Kusnita Yusmiarti (2016). *Deployment diagram* digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem. *Deployment diagram* digunakan untuk memvisualisasikan, menspesifikasikan, dan mendokumentasikan proses yang terjadi pada suatu sistem perangkat lunak berbasis *Object Oriented* yang akan dibangun. Tujuan dan fungsi dari *deployment diagram* yaitu untuk menggambarkan secara umum proses yang terjadi pada suatu *sistem/software*. *Node* dalam *UML* merepresentasikan *hardware* atau *software execution environments*. *Node* bisa terhubung melalui *communication path* untuk membuat *system jaringan* dari *arbitrary complexity*. *Artifacts* dalam *UML* mempresentasikan spesifikasi dari bentuk *physic* informasi yang digunakan atau dihasilkan *development process* di tunjukkan pada Tabel 2.6 *Deployment Diagram*:

Tabel 2.6 *Deployment Diagram*

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

Contoh *Deployment Diagram*



Gambar 2.8 Contoh *Deployment Diagram*

Pada Gambar 2.8 *website* pemesanan ruang rapat Diskominfo Statistik dan Persandian Surakarta terbagi menjadi dua node yaitu *pegawai* dan *System*, pada node *pegawai* terdapat dua *component* yaitu *website* pemesanan ruang rapat Diskominfo Statistik dan Persandian Surakarta dan *database* ruang rapat, sedangkan *system* hanya memiliki satu *component* yaitu *interfacewebsite* pemesanan ruang rapat Diskominfo Statistik dan Persandian Surakarta.

2.3.13 *StarUML*

Menurut Made (2015), *StarUML* merupakan proyek *open source* yang digunakan untuk mengembangkan *platform Unified Modeling Language (UML)* atau *Model Driven Architecture* yang cepat, fleksibel, dapat diperluas, memiliki banyak fitur, dan gratis. *StarUML* dikembangkan dalam bahasa pemrograman *Delphi*. *StarUML* merupakan proyek yang *multi-lingual* dan tidak bergantung pada bahasa pemrograman yang spesifik, sehingga bahasa pemrograman apapun dapat digunakan untuk mengembangkan *StarUML*, seperti *C/C++*, *Java*, *Visual Basic*, *Delphi*, *Jscript*, *VBScript*, *C#*, *VB.NET*, dan sebagainya.

2.3.14 *Sublime Text*

Menurut Faridl (2015), *Sublime text* adalah teks *editor* berbasis *Python*, sebuah teks *editor* yang elegan, kaya fitur, *crossplatform*, mudah dan simpel yang cukup terkenal di kalangan *developer* (pengembang), penulis dan *desainer*. Para *programmer* biasanya menggunakan *sublimetext* untuk menyunting *sourcecode* yang sedang ia kerjakan. Sampai saat ini *sublimetext* sudah mencapai versi 3

Sublimetext mempunyai beberapa keunggulan-keunggulan yang dapat membantu pengguna dalam membuat sebuah *webdevelopment*.

Berikut beberapa fitur yang diunggulkan dari aplikasi *Sublime Text*:

1. *Goto Anything*

Fitur yang sangat membantu dalam membuka file ataupun menjelajahi isi dari file hanya dengan beberapa keystrokes.

2. *Multiple Selections*

Fitur ini memungkinkan user untuk mengubah secara interaktif banyak baris sekaligus, mengubah nama variabel dengan mudah, dan memanipulasi file lebih cepat dari sebelumnya.

3. *Command Pallete*

Dengan hanya beberapa keystrokes, user dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalui menu.

4. *Distraction Free Mode*

Bila user memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu user dengan memberikan tampilan layar penuh.

5. *Split Editing*

Dapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan editing perpecahan. Mengedit sisi file dengan sisi, atau mengedit dua lokasi di satu file. Dapat mengedit dengan banyak baris dan kolom yang user inginkan.

6. *Instant Project Switch*

Menangkap semua file yang dimasukkan kedalam project pada aplikasi ini. Terintegrasi dengan fitur *Goto Anything* untuk menjelajahi semua file yang ada ataupun untuk beralih ke file dalam *project* lainnya dengan cepat.

7. *Plugin API*

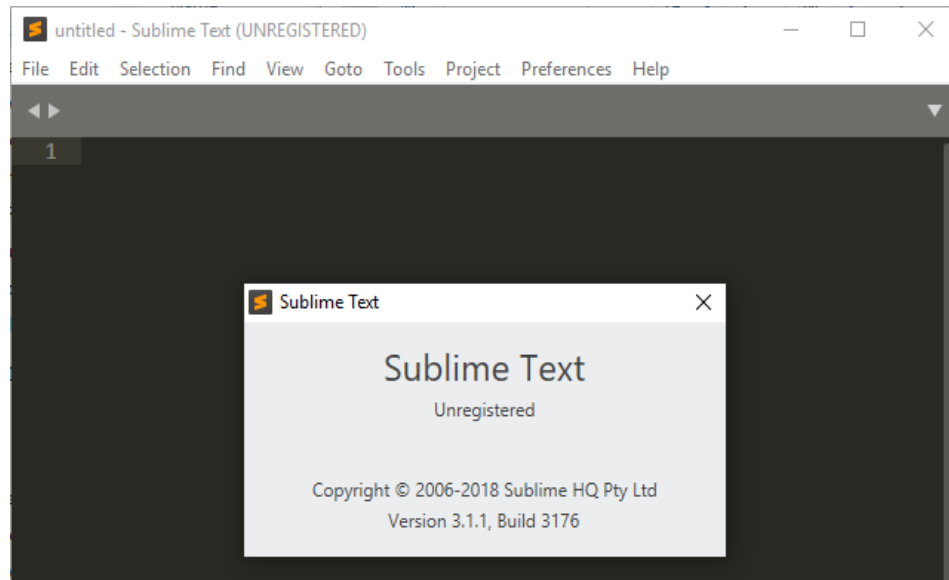
Dilengkapi dengan *plugin API* berbasis *Python* sehingga membuat aplikasi ini sangat tangguh.

8. *Customize Anything*

Aplikasi ini memberikan user fleksibilitas dalam hal pengaturan fungsional dalam aplikasi ini.

9. *Cross Platform*

Aplikasi ini dapat berjalan hampir disemua *operating system modern* seperti *Windows, OS X, dan Linux based operating system*.



Gambar. 2.9 Contoh *Sublime Text*

2.3.15 Pengujian Sistem

Pengujian sistem menggunakan 2 metode, yaitu:

1. *Black Box Testing*

Menurut Shalahuddin dan Rosa (2015), *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 dibuat untuk melakukan pengujian *black-box testing* harus dibuat dengan kasus benar dan kasus salah.

2. *Survey Kepuasan Pengguna*

Menurut Sugiyono (2017), Kuesioner adalah Teknik pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk dijawabnya. Kuesioner merupakan teknik

pengumpulan data yang efisien bila peneliti tahu dengan pasti variabel yang akan diukur dan tahu apa yang bisa diharapkan dari responden. Selain itu, kuesioner juga cocok digunakan bila jumlah responden cukup besar dan tersebar di wilayah yang luas.