

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Berdasarkan penelitian yang dilakukan oleh Muhammad Soleh Ferdiansyah, Mohammad Jasri, dan Widjianto yang berjudul “Aplikasi *Quick Response* Dalam Melayanai Pengaduan Kerusakan Sarana STT Nurul Jadid Berbasis Android dan Web” pada tahun 2016. Dalam penelitian tersebut, STT Nurul Jadid membahas mengenai pola yang diterapkan pengelolaan pengaduan yang diajukan civitas akademika serta memaparkan beberapa contoh yang ditempuh dengan tujuan pengelolaan pengaduan dapat menjadi pembanding yang baik terhadap instansi lain. Pengelolaan Pengaduan menggunakan metode deskriptif eksploratif terhadap peraturan yang telah ada. Metode tersebut digunakan untuk menjelaskan secara lugas dan terstruktur terhadap pengaduan yang disampaikan. Untuk merealisasikan metode tersebut, STT Nurul Jadid telah membuat sebuah sistem pelayanan penyediaan informasi dan pengaduan menggunakan sarana teknologi website. Untuk menunjang keberhasilan pelayanannya, STT Nurul Jadid telah memberikan aturan yang jelas mengenai pemberian informasi pengaduan dan melakukan transparansi pada data pengaduan serta manajemen yang sederhana.

Pada penelitian Aplikasi *Quick Response* Dalam Melayanai Pengaduan Kerusakan Sarana STT Nurul Jadid Berbasis Web yang dilakukan oleh Muhammad Soleh Ferdiansyah, Mohammad Jasri, dan Widjianto mempunyai kesamaan dengan peneliti yaitu sama-sama bertujuan untuk membuat pengguna (mahasiswa atau masyarakat) dapat menjelaskan secara jelas dan transparan apa saja keluhan yang dapat dilaporkan.

Kedua, penelitian yang dilakukan oleh Abdi Arrafi dan Harsiti di tahun 2014 yang berjudul “Perancangan Aplikasi Pengelolaan Pengaduan Masyarakat PNPM Mandiri Perkotaan Di Provinsi Banten Dengan Metode Waterfall”. Penelitian tersebut bertujuan untuk menyajikan informasi yang terkait dengan kendala teknis dan permasalahan yang terjadi di lapangan. Serta memberikan hal

yang mudah dalam pengelolaan laporan pengaduan dan pembuatan laporan sesuai dengan kebutuhan Pengguna. Selain itu sistem yang telah terkomputerisasi terhadap program komputer dapat membantu mempermudah dalam menyelesaikan sebuah masalah.

Aplikasi tersebut didesain bertujuan supaya mempermudah hal dalam proses pencatatan gangguan dan dalam hal penyimpanan data, sehingga proses pengelolaan pengaduan dapat dioptimalkan dengan lebih cepat dan lebih tepat tentunya. Dari hasil penelitian yang dilakukan oleh Abdi Arrafi dan Harsiti dengan peneliti terdapat juga beberapa kesamaan yaitu sama-sama menggunakan media website dalam mengelola semua laporan pengaduan atau keluhan, mempermudah pengguna(masyarakat) dalam menyampaikan aspirasi mereka berupa keluhan maupun saran, serta memiliki tujuan untuk memonitor jalannya proses pengelolaan pengaduan tersebut dengan jelas.

Ketiga, penelitian yang dilakukan oleh Arin Nurhita Hapsari dan Rini Rachmawati di tahun 2017 yang berjudul “ Pemanfaatan Aplikasi Lapor Sleman Sebagai Pelayanan Pengaduan Di Kabupaten Sleman”.

Berdasarkan penelitian ketiga yang dilakukan oleh Arin Nurhita Hapsari dan Rini Rachmawati, penelitian tersebut bertujuan supaya masyarakat lebih mudah dalam memberikan pengaduan. Lapor Sleman ini dapat diakses melalui *website* yang dapat digunakan di mana saja dan kapan saja. Hal ini tentu mempermudah penyampaian informasi serta pengaduan dari masyarakat terhadap pemerintah Kabupaten Sleman. Selain itu aplikasi ini lebih efektif dan efisien karena seluruh laporan yang ada telah terhubung pada instansi yang bersangkutan. Manfaat yang didapat dari aplikasi ini yaitu semua laporan menjadi bersifat transparan sehingga dapat dilihat oleh pengguna aplikasi. Selain itu pengguna juga dapat mengetahui kondisi di Kabupaten Sleman itu sendiri, serta hal ini juga bisa digunakan sebagai fungsi kontrol terhadap pemerintah terkait laporan pengguna yang tidak kunjung diselesaikan atau direspon. Disisi lain dengan adanya timbal balik dari pemerintah yang baik membuat seluruh masyarakat merasa lebih mudah untuk menyampaikan keluhan, pengaduan dan saran .

Keempat, penelitian yang dilakukan oleh Iqbal Kurniawan di tahun 2017 yang berjudul “Layanan Pengaduan Berbasis Web Pada Dinas Perumahan Dan Kawasan Pemukiman Kota Metro”. Penelitian tersebut bertujuan untuk membuat layanan pengaduan berbasis *web* yang dapat diakses oleh Kepala Bidang dan Masyarakat darimana saja, sehingga pengaduan dapat dengan cepat ditangani serta memudahkan staf Bidang untuk membuat laporan pengaduan karena tidak perlu mengetik ulang pengaduan. Selain itu dapat disimpulkan dari pembuatan layanan pengaduan berbasis web ini adalah tersedianya alternatif untuk masyarakat dalam melakukan pengaduan, dan memudahkan Dinas Perumahan dan Kawasan Pemukiman Kota Metro dalam memproses dan melayani pengaduan masyarakat.

Kelima, penelitian yang dilakukan oleh Adian Fathur Rochim dan Maman Somantri di tahun 2012 yang berjudul “Sistem Aplikasi Pencatatan Tindak Kejahatan Pada Polsek Tegal Selatan Berbasis Web”. Penelitian tersebut bertujuan untuk membantu petugas kepolisian dalam melakukan pencatatan tindak kejahatan serta kriminal sehingga diharapkan pelayanan yang diberikan dapat lebih cepat serta memudahkan petugas kepolisian untuk mengetahui tingkat kriminalitas yang ada di wilayahnya dan memberikan informasi tersebut ke masyarakat. Sistem Informasi ini dapat mempermudah pengguna khususnya pihak kepolisian dalam mengerjakan serta melakukan pendataan kasus-kasus kejahatan yang pernah terjadi di kelurahan khususnya wilayah tegal selatan.

Dari hasil penelitian yang dilakukan oleh Adian Fathur Rochim dan Maman Somantri terdapat beberapa kesamaan yaitu pengelolaan pengaduan sama-sama menggunakan media *website*, serta memiliki kesamaan pada proses tindak lanjut yang dilakukan oleh instansi terkait.

Dari kelima penelitian sebelumnya, dapat disimpulkan bahwa fungsi pembangunan sistem penanganan keluhan bertujuan untuk mempermudah pengguna yaitu masyarakat atau mahasiswa dalam menyampaikan aspirasinya dengan jelas dan dapat ditindaklanjuti oleh pihak yang berwenang dengan pengelolaan yang transparan secara terkontrol.

Berdasarkan dari kelima penelitian yang sudah dilakukan diatas, maka dapat disimpulkan peneliti akan membuat fitur sebagai berikut:

1. Sistem akan menghasilkan keluaran (*output*) berupa data *print* serta bisa juga dijadikan dalam bentuk bukti *pdf* yang mana dapat mempermudah petugas terkait dalam menindaklanjuti laporan atas pengaduan pengguna.
2. Menggunakan MySql sebagai media penyimpanan data.
3. Website dibuat menggunakan PHP dengan terdapat tiga pengguna yaitu Super Admin, Petugas, dan User.
4. User tidak perlu melakukan login untuk menyampaikan aspirasi keluhan gangguan yang ada.

Melalui sistem ini, pengguna atau masyarakat bisa menyampaikan aspirasi mereka berupa keluhan atau pengaduan. Untuk menggunakan website ini, pengguna atau masyarakat akan mendapatkan akses dengan sebuah *Nomor Unik* sebagai tanda pengguna yang diberikan oleh petugas terkait dan kemudian memasukan *Nomor Unik* tersebut ke dalam *web* layanan untuk menyampaikan aspirasi berupa keluhan, pengaduan, dan kritik yang kemudian akan ditindaklanjuti oleh petugas terkait untuk segera diselesaikan.

2.2 Landasan Teori

2.2.1 Pengetian Sistem

Pengetian dari suatu sistem adalah sebuah kesatuan dari bagian-bagian atau unsur-unsur yang mengandung arti berbeda-beda yang saling berhubungan satu dengan yang lain, yang mana hal-hal tersebut dijalankan sesuai skema yang kompleks untuk mencapai suatu tujuan.

Pengertian Sistem menurut yang telah dikemukakan oleh para ahli adalah sebagai berikut:

Dalam buku yang ditulis oleh Sutarman (2009:5) yang berjudul *Pengantar Teknologi Informasi* : “ Sistem adalah kumpulan elemen yang saling berinteraksi dalam suatu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama ”.

Dalam buku yang ditulis oleh Mulyadi (2010:5) yang berjudul *Sistem Akuntansi* menyatakan bahwa “ Sistem adalah jaringan prosedur

yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan-kegiatan pokok perusahaan sedangkan prosedur adalah suatu urutan kegiatan klerikal, biasanya melibatkan beberapa orang dalam satu departemen atau lebih yang dibuat untuk menjamin penanganan secara seragam transaksi perusahaan yang terjadi secara berulang-ulang.”

Dalam buku yang ditulis oleh Jogiyanto (2009:34) dengan judul *Analisis dan Desain Sistem Informasi* : “Sistem dapat di definisikan dengan pendekatan prosedur dan dengan pendekatan komponen ”. Dimana sistem dan prosedur adalah sebuah kesatuan yang tidak bisa dipisahkan satu sama lain. Suatu sistem baru, hanya bisa terbentuk jika di dalamnya terdapat prosedur-prosedur yang mengikutinya.

Dalam buku yang ditulis oleh Azhar Susanto (2013:22) yang berjudul *Sistem Informasi Akuntansi* : “ Sistem adalah kumpulan atau group dari sub sistem atau komponen apapun baik phisik ataupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu”.

Dalam pendekatan untuk mendefinisikan suatu sistem terdapat dua kelompok dasar yaitu sebagai berikut:

1. Pendekatan sistem berdasarkan prosedur

Sistem merupakan gabungan antara jaringan dan prosedur yang saling berhubungan satu sama lain, serta saling bekerjasama guna melakukan suatu pekerjaan maupun menyelesaikan permasalahan tertentu.

2. Pendekatan sistem berdasarkan komponen

Sistem merupakan kumpulan dari bagian-bagian elemen yang saling berkaitan secara teratur dimana hal tersebut akan membentuk suatu totalitas yang dapat menyelesaikan permasalahan tertentu.

Menurut pendapat beberapa ahli yang telah dikemukakan diatas maka secara tidak langsung dapat diambil kesimpulan bahwa suatu sistem merupakan kumpulan dari beberapa bagian atau sub

sistem yang dibentuk menjadi satu kesatuan dan didesain untuk mencapai tujuan tertentu.

a. Karakteristik Sistem

Dalam buku yang ditulis oleh Agus Mulyanto (2009:2), dengan judul *Sistem Informasi Konsep dan Aplikasi* tertulis sebagai berikut seperti:

1. Komponen Sistem (*Components System*)

Suatu sistem tersusun dari komponen-komponen yang saling berinteraksi dan saling bekerja sama untuk membentuk satu kesatuan. Selain itu, sebuah sistem akan terletak pada lingkungan yang tidak kosong, dengan kata lain sebuah sistem akan terletak dan memiliki fungsi di dalam lingkungannya yang mana lingkungan tersebut terdapat sistem lainnya. Suatu sistem yang memiliki komponen dari sistem lain yang lebih besar, hal tersebut dapat disebut juga dengan *subsistem*, disisi lain sistem yang lebih besar tersebut dapat dikatakan sebagai lingkungannya.

2. Batasan Sistem (*Boundary*)

Batasan sistem merupakan sebuah dinding penghalang atau dapat dikatakan sebagai sebuah pemisah antara sistem satu dengan sistem yang lainnya atau suatu sistem dengan lingkungan luarnya.

3. Lingkungan (*Environment*)

Lingkungan luar merupakan sekuruh hal yang berada di luar batas dari sistem yang mana hal itu mampu memberikan pengaruh terhadap operasi sebuah sistem, baik dalam pengaruh yang memberikan dampak positif maupun dampak negatif. Pengaruh yang positif tersebut tentu harus senantiasa dijaga dan dipertahankan dimana hal tersebut tentu dapat mendukung kelangsungan suatu operasi dari

sebuah sistem yang dijalankan. Kemudian dari segi lingkungan yang memberikan dampak negatif tentu perlu tindakan pengendalian serta hal tersebut jangan dibiarkan supaya tidak menghambat maupun merugikan kelangsungan dari sebuah sistem itu sendiri.

4. Penghubung (*Interface*) Komponen

Penghubung antar komponen merupakan suatu media penghubung atau suatu hal yang menjembatani antara suatu subsistem dengan subsistem lainnya. Penghubung tersebut digunakan menjadi sebuah media untuk melakukan sebuah kegiatan yang dimulai dari data masukan (*input*) sampai data keluaran (*output*). Manfaat dari adanya penghubung dari suatu subsistem tersebut yaitu dapat melakukan interaksi dan integrasi dengan subsistem yang lainnya untuk membentuk menjadi satu kesatuan.

5. Masukan (*Input*)

Sebuah masukan atau dapat juga disebut input merupakan suatu data yang dimasukkan ke dalam suatu sistem. Contoh dari masukan atau *input* tersebut dapat berupa masukan perawatan (*maintenance input*), yaitu data yang dimasukkan supaya sistem tersebut bisa beroperasi dengan baik. Kemudian contoh yang kedua dapat berupa masukan dalam bentuk sinyal (*signal input*), yaitu sebuah masukan dimana hal tersebut akan diproses untuk mendapatkan hasil keluaran.

6. Pengolahan (*Processing*)

Pengolahan (*process*) yaitu bagian yang melaksanakan sebuah proses atau cara yang dilakukan dengan tujuan untuk memberi perubahan bentuk mulai dari

masuk hingga menghasilkan keluaran seperti yang diinginkan.

7. Sasaran (*Objective*) dan Tujuan (*Goal*)

Sasaran dan tujuan menjadi hal yang membuat operasi sistem menjadi terarah dan terkendali. Apabila suatu operasi sistem tidak memiliki sebuah sasaran dan tujuan, maka operasi sistem tersebut hanya dapat menjadi hal yang tidak memiliki nilai guna. Karena sasaran dan tujuan inilah yang dapat dijadikan sebagai arahan dari suatu sistem itu sendiri. Tanpa adanya sasaran dan tujuan dalam operasi sistem, maka sistem akan berjalan menjadi tidak terarah dan tidak terkendali.

8. Keluaran (*Output*)

Keluaran (*output*) adalah hasil dari sebuah pemrosesan yang diawali dengan memasukan data input hingga menghasilkan suatu keluaran atau data keluaran. Contoh dari keluaran dapat berupa data atau informasi yang dapat digunakan untuk masukan pada sistem lain atau hanya sebagai sisa pembuangan.

9. Umpan Balik (*Feed Back*)

Umpan balik yaitu sebuah proses dimana sebagian dari keluaran (*output*) di umpan balikan ke bagian masukan (*input*). Hal ini sering dipakai sebagai pengendalian yang dilakukan oleh bagian kendali sistem guna mengecek hal-hal yang memungkinkan terjadinya penyimpangan proses dalam sistem serta dapat mengembalikannya ke dalam kondisi yang normal.

2.2.2 Pengertian Keluhan(*complain*)

Keluhan pada dasarnya merupakan sebuah bagian dari bentuk komunikasi dari sebuah informasi yang memiliki unsur ketidaksesuaian

yang dirasakan oleh pihak penerima sebuah layanan jasa atau sebuah produk (James, 2006:20). Oleh karena itu, suatu keluhan itu merupakan hal yang sangat penting dan sangat dibutuhkan bagi pihak penyedia layanan maupun penerima layanan. Karena berawal dari suatu keluhan akan menjadikan sebuah informasi, baik informasi positif atau informasi negatif. Bahkan keluhan dapat menjadi sebuah komunikasi aktif antara penyedia dan penerima layanan yang akan menjurus pada sebuah interaksi.

Keluhan dari suatu pelanggan menjadi sebuah ungkapan atas ketidakpuasan dari seorang pelanggan atau pengguna dari suatu produk yang berbentuk barang maupun jasa. Namun tidak semua keluhan pelanggan dapat ditangkap dengan jelas oleh penyedia barang atau jasa. Dalam hal ini keluhan dapat berdampak langsung dan tidak langsung terhadap kinerja perusahaan. Akan tetapi kebanyakan kasus tersebut menimbulkan dampak yang tidak dapat dirasakan secara langsung oleh perusahaan atau penyedia. Oleh karena itu, sebuah perusahaan atau penyedia harus memiliki kemampuan dalam mendeteksi keluhan dari seorang pelanggan untuk memperbaiki serta mengevaluasi hal-hal yang menunjang perusahaan.

2.2.3 Pengertian Informasi

Menurut Jogiyanto HM (1999: 692), “Informasi didefinisikan sebagai hasil dari suatu pengolahan data dalam bentuk yang lebih berguna serta lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan sebuah keputusan”

Menurut Kadir (2008:3), “Informasi adalah data yang telah diolah menjadi bentuk bermakna dan berguna bagi manusia. Informasi dapat juga diartikan menjadi data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakannya.” Informasi dapat disimpulkan menjadi :

1. Informasi bersumber pada data.

2. Memberikan suatu nilai tambah atau pengetahuan bagi yang menggunakan.
3. Dapat digunakan untuk pengambilan keputusan.

2.2.4 Pengertian Sistem Informasi

Informasi dapat diperoleh dari sistem informasi (*Information System*) atau disebut juga dengan *Processing Systems* atau *Information Generating Systems*.

Menurut pendapat O'brian (2012:16), mengatakan bahwa “sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi.”

Menurut pendapat Agus Mulyanto (2009:29), mengatakan bahwa “dalam mendefinisikan sistem informasi sebagai suatu komponen yang terdiri dari manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai suatu tujuan.”

Menurut pendapat Alter dalam Kadir (2009:7), mengatakan bahwa “sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.”

Berdasarkan pendapat para ahli yang telah dikemukakan di atas, maka dapat disimpulkan bahwa sebuah sistem informasi memiliki sebuah peranan yang sangat penting dalam menyampaikan data maupun informasi yang berguna bagi penggunanya.

2.3 Teknologi Pengembangan Aplikasi

2.3.1 UML(*Unified Modeling Language*)

Unified Modeling Language (UML) merupakan standar bahasa yang banyak digunakan di dunia industri pada saat ini untuk mengartikan sebuah *requirement*, membuat sebuah analisa dan desain serta untuk

mendefinisikan gambaran arsitektur dalam program yang berorientasi pada objek atau biasa disebut OOP (Nugroho,2010:6). Penggunaan model *Unified Modeling Language* (UML) bertujuan untuk mengidentifikasi bagian-bagian mana saja yang termasuk dalam lingkup sistem di dalam sebuah aplikasi. Model UML yang dipakai dalam pembuatan Aplikasi Pengaduan Gangguan LPSE Berbasis Web antara lain adalah *Use Case Diagram*, *Activity Diagram*, *Class Diagram*, *ER Diagram*. Bagian dari UML antara lain yaitu:

a. *Use Case Diagram*

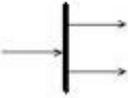
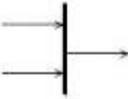
Uses case diagram menjelaskan tentang sebuah hubungan antara satu actor maupun lebih dengan menggunakan rancangan sistem yang telah dibuat. Diagram tersebut dapat digunakan untuk mengetahui fungsi yang telah ada di dalam sistem serta pihak mana saja yang memiliki hak untuk menggunakan fungsi yang dibutuhkan. Berikut beberapa simbol yang digunakan dalam *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 eksplisit.
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.

Gambar 2.1 Struktur Use Case Diagram

b. *Activity Diagram*

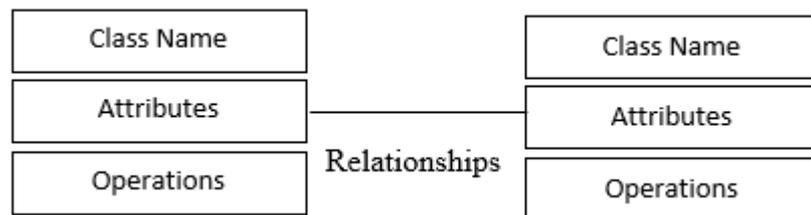
Activity Diagram merupakan diagram yang digunakan untuk menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*.

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)

Gambar 2.2 Struktur Activity Diagram

c. *Class Diagram*

Class Diagram merupakan sebuah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem meliputi : Nama Kelas (*ClassName*), Atribut (*Attributes*), Operasi (*Operations*), dan Relasi (*Relationships*).



Gambar 2.3 Struktur Class Diagram

Adapun dalam suatu diagram kelas atribut dan method dapat memiliki salah satu sifat berikut :

1. Private, yaitu hanya bisa dipanggil dari dalam kelas itu sendiri atau tidak dapat dipanggil dari luar kelas yang bersangkutan. Methode atau atribut ini diawali dengan tanda “-“.
2. Protected, yaitu hanya dapat dipanggil oleh kelas yang bersangkutan dan kelas turunannya. Methode diawali dengan tanda “#”.
3. Public, dapat dipanggil dari semua objek. Methode atau atribut diawali dengan tanda “+”

d. *ER Diagram*

Entity Relationship Diagram (ERD) merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD digunakan untuk memodelkan struktur data dan hubungan antar data.

2.4 Model Pengembangan Aplikasi

2.4.1 SDLC (*Software Development Life Cycle*)

Metode Pengembangan SDLC (*System Development Life Cycle*) merupakan suatu bentuk yang digunakan untuk menggambarkan tahapan dan langkah-langkah didalam tahapan tersebut dalam proses pengembangannya. Model SDLC yang dipakai dalam pengembangan aplikasi adalah model *Waterfall*. Menurut (Sommerville, 2011), model

Waterfall adalah sebuah contoh dari proses perencanaan dimana semua proses kegiatan harus terlebih dahulu direncanakan dan dijadwalkan sebelum dikerjakan. *Waterfall Model* atau *Classic Life Cycle* merupakan model yang paling banyak dipakai dalam *Software Engineering (SE)*. Disebut *Waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

Berikut tahapan-tahapan tersebut terdiri atas:

a. Kebijakan Sistem dan Perencanaan Sistem

Kebijakan sistem (*system policy*) adalah sebuah aturan yang bersifat mengikat dan dijadikan sebagai landasan serta dukungan dari manajemen puncak dalam membuat sebuah perencanaan sistem. Perencanaan sistem (*system planning*) merupakan pedoman yang digunakan untuk melakukan pengembangan dari suatu sistem.

b. Analisa Sistem

Tahapan ini merupakan tahap dalam menganalisa suatu sistem untuk menemukan kekurangan atau kelemahan yang ada pada sistem tersebut sehingga dapat dilakukan perbaikan untuk ke depannya.

c. Desain Sistem dalam Arti Umum

Desain sistem dalam arti umum merupakan suatu persiapan sebelum melakukan desain terinci, yang mana dalam hal tersebut menggambarkan seluruh komponen sistem informasi yang akan didesain secara rinci.

d. Desain Sistem Terinci

Desain sistem secara terinci yaitu ditujukan kepada sistem pemrograman sebuah komputer dan ahli teknik yang akan diimplementasikan.

e. Seleksi Sistem

Seleksi sistem merupakan tahapan dimana dilakukan pemilihan perangkat keras maupun perangkat lunak untuk menunjang sistem informasi.

f. Implementasi

Implementasi sistem adalah tahapan dimana dilakukan penerapan sebuah sistem dengan tujuan agar system tersebut siap untuk dioperasikan.

g. Perawatan Sistem

Perawatan sistem adalah tahapan terakhir dimana telah selesainya pengembang sistem atau telah selesainya sistem yang dioperasikan.

2.5 Aplikasi Berbasis Web

Aplikasi Berbasis *Web* merupakan contoh sebuah aplikasi yang dapat digunakan melalui akses dengan internet, yang mana pada era sekarang lebih banyak digunakan serta lebih luas dalam hal penggunaannya. Kebanyakan perusahaan yang berkembang menggunakan Aplikasi Berbasis *Web* untuk merencanakan SDM (sumber daya manusia) mereka serta digunakan untuk mengelola perusahaan terkait baik secara langsung maupun tidak langsung.

Menurut pendapat ahli web adalah salah satu contoh sumber daya dari internet yang berkembang dengan sangat pesat(Kadir 2003). Penyajian informasi web menggunakan pendekatan melalui hyperlink, yang mana hal tersebut memungkinkan gambar, teks maupun objek lain menjadi sebuah acuan untuk mengakses halaman yang lainnya. Dengan menggunakan pendekatan ini seorang pengguna dapat memperoleh informasi dengan beranjak dari satu halaman ke halaman yang lainnya.

Aplikasi Berbasis *Web* ini juga bisa digunakan untuk berbagai macam tujuan yang mana intinya untuk memudahkan suatu hal dalam pekerjaan. Salah satu contohnya yaitu, Aplikasi Berbasis *Web* digunakan untuk membuat sebuah struk atau nota yang mana data informasi yang ditampilkan tersebut sebelumnya sudah disimpan di dalam database. Selain itu, Aplikasi ini juga dapat digunakan untuk

mengatur jumlah persediaan barang yang ada, yang mana fitur tersebut sangat berguna bagi mereka yang melakukan usaha jual beli barang secara online.

Selain fungsi yang ada tersebut, keunggulan yang sangat bagus dari Aplikasi Berbasis *Web* yaitu aplikasi tersebut sangatlah enteng serta ringan dan dapat diakses dengan cepat oleh pengguna dengan melalui browser yang telah terkoneksi internet. Yang mana pengguna dalam hal ini dapat mengakses data atau informasi yang mereka butuhkan melalui *smartphone*, *tablet*, *laptop* dan komputer *PC* dengan sangat mudah. Hal ini tidak seperti aplikasi-aplikasi *desktop* yang mana pengguna harus menginstal perangkat lunak atau aplikasi yang diperlukan hanya untuk mengakses data atau informasi yang mereka butuhkan.

Aplikasi *Web* dapat dibagi menjadi dua jenis yaitu :

1. Aplikasi *Web* Statis

Web statis adalah *web* yang dibentuk dengan menggunakan *HTML* dan *web* yang dimana penggunaanya tidak dapat mengubah *website* secara langsung melalui browser. Sehingga *web* statis ini hanya dapat diubah secara manual saja. Kekurangan aplikasi *web* ini terletak pada pemeliharaan program yang dilakukan secara terus menerus untuk mengikuti perkembangan yang terus terjadi.

2. Aplikasi *Web* Dinamis

Web dinamis adalah *web* dimana terjadi suatu perubahan informasi pada halaman *web*, yang dilakukan dengan tanpa adanya perubahan pada program tersebut. Akan tetapi hal ini hanya ada perubahan datanya saja. Sebagai contoh, perubahan informasi dapat dilakukan oleh administrator secara berkala dengan mudah dengan dikoneksikan ke basis data.

2.6 Konsep Basis Data

2.6.1 Definisi Database

Berikut definisi-definisi Database menurut para ahli:

“*Database* adalah sekumpulan file yang saling berhubungan dan terorganisasi atau kumpulan *record-record* yang menyimpan data dan hubungan diantaranya”, Sutarman (2012:15).

“*Database* adalah sekumpulan data store (bisa dalam jumlah yang sangat besar) yang tersimpan dalam *magnetic disk, optical disk, magnetic drum*, atau media penyimpanan sekunder lainnya”, Ladjamudin A (2013).

Menurut Edy Suthanta (2014) dalam bukunya yang berjudul *Analisa Sistem Basis Data*, “Basis data dapat dipahami sebagai suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkaan data, data disimpan dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali, data dapat digunakan satu atau lebih program-program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan dengan program yang akan menggunakannya, data disimpan dengan sedemikian rupa sehingga proses penambahan, pengembalian, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.”

Dari pengertian diatas dapat disimpulkan bahwa *Database* adalah sekumpulan file yang saling berhubungan yang menyimpan data dan tersimpan dalam sebuah media penyimpanan.

Sistem *Database* memiliki enam komponen penting:

1. *Hardware*

Hardware adalah perangkat keras dimana dapat berupa perangkat komputer, media penyimpanan dan media untuk komunikasi dalam sistem jaringan.

2. *Operating System*

Operating System adalah perangkat lunak dimana perangkat inilah yang memberikan fungsi kontrol untuk mengendalikan seluruhnya termasuk sumber daya, serta dalam hal untuk melakukan operasi dasar pada sistem komputer.

3. *Database*

Database adalah suatu basis data yang mana hal tersebut telah mewakili sistem tertentu untuk dikelola. Suatu *database* bisa terdiri dari satu basis data maupun lebih dari satu basis data.

4. *DBMS (Database Management System)*

Database Management System merupakan perangkat lunak yang digunakan untuk mengelola basis data. Contoh dari *Database Management System* dalam kelas sederhana seperti : dBase, Foxbase, Rbase, MS. Access. Sedangkan contoh *Database Management System* dalam kelas kompleks seperti : MS. SQL Server, Oracle, Sybase.

5. *User*

User merupakan sebutan untuk para pengguna sistem basis data, yaitu orang-orang yang saling melakukan interaksi satu sama lain menggunakan sistem basis data, mulai dari kegiatan merancang hingga para pengguna pada tingkat akhir.

6. *Optional Software*

Optional software menjadi salah satu pelengkap untuk perangkat lunak yang mendukung sistem. *Software* ini memiliki sifat opsional atau pilihan yaitu sesuatu hal yang dapat dilakukan ataupun tidak dilakukan, diambil atau tidak diambil yang sifatnya tidak wajib.

2.6.2 Definisi MySql

MySQL (*My Structure Query Language*) adalah sebuah program pembuat *database* yang bersifat *open source* (Bunafit Nugroho, 2004).

Dengan bahasa perintah terstruktur yang distandarisasikan untuk semua program pengakses database seperti Oracle, Postgres, SQL, SQL Server, dll.

Dalam hal penggunaan SQL dibagi menjadi tiga sub perintah kategori, yang diantaranya yaitu *Data Definition Language*, *Data Manipulation Language* dan *Data Control Language*.

1. Pengertian *Data Definition Language*

Data Definition Language adalah sebuah perintah SQL yang digunakan untuk mendefinisikan, menciptakan, dan menghapus sebuah database ataupun tabel. Perintah-perintah DDL diantaranya yaitu *CREATE*, *DROP*, dan *ALTER*.

- a. *Create* : perintah yang berfungsi sebagai pembuatan suatu data, termasuk untuk pembuatan database baru, table yang baru, serta kolom baru.
- b. *Alter* : perintah ini berfungsi untuk melakukan perubahan atau pengaturan struktur tabel yang sebelumnya telah dibuat. Tugasnya dapat berupa penggantian nama tabel, penambahan jumlah kolom, pengubahan kolom, penghapusan kolom, maupun memberikan atribut pada kolom.
- c. *Drop* : perintah yang memiliki fungsi untuk melakukan penghapusan database dan tabel.

2. Pengertian *Data Manipulation Language*

Data Manipulation Language yaitu sebuah bahasa yang digunakan untuk mengubah atau memanipulasi isi pada sebuah tabel.

- a. *Insert* : perintah yang berfungsi untuk menyisipkan atau memasukkan data baru ke dalam tabel. Penggunaannya setelah database dan table telah dibuat.
- b. *Update* : perintah yang berfungsi untuk mengubah atau memperbarui data yang lama menjadi data baru.

c. *Delete* : perintah yang berfungsi untuk menghapus data dari tabel. Data yang sudah terhapus tidak akan dapat dikembalikan seperti keadaan semula. Karena ketika dilakukan penghapusan suatu data, perintah yang telah dijalankan tidak dapat digagalkan.

3. Pengertian *Data Control Language*

Data Control Language yaitu sebuah perintah yang berhubungan dengan pengendalian akses ke database. Perintah dalam *Data Control Language* diantaranya seperti :

- a. *Grant* : suatu perintah yang digunakan sebagai pemberian hak/izin akses dari pemilik utama (*administrator*) server kepada pengguna biasa (*user*). Hak/izin akses tersebut dapat berupa hak/izin untuk membuat (*Create*), mengambil (*Select*), menghapus (*Delete*), mengubah (*Update*), serta hak/izin khusus yang berkaitan pada sistem *database*.
- b. *Revoke* : suatu perintah yang merupakan kebalikan dari fungsi *Grant*, yaitu penghilangan atau pencabutan hak/izin akses yang sebelumnya diberikan oleh *administrator* kepada *user*.

2.7 Bahasa Pemrograman

2.7.1 HTML (*Hypertext Markup Language*)

HTML (*Hypertext Markup Language*) adalah bahasa pemrograman yang dirancang untuk memungkinkan pembuatan situs web. Dokumen HTML merupakan dokumen yang disajikan pada *web browser*.

<body>...</body>	untuk tampilan halaman web
 	untuk pindah ke baris berikutnya
<h1>...</h1> ... <h6>...</h6>	untuk membuat heading. Mempunyai 6 jenis heading <h1>...</h1> merupakan heading yang paling besar sampai <h6>...</h6>, heading yang paling kecil
<hr>	membuat garis horizontal
<center>...</center>	membuat text ditengah
<title>...</title>	menulis judul
...	membuat huruf tebal
<big>...</big>	memperbesar teks dari ukuran normal
<i>...</i>	membuat huruf miring
<small>...</small>	memperkecil teks dari ukuran normal
_{...}	membuat huruf subskrip
^{...}	membuat huruf superskrip
<u>...</u>	membuat garis bawah
...	membuat list berupa bullet
...	membuat list berupa nomor
	membuat item-item pada list
	menampilkan gambar

Gambar 2.4 Simbol Tag HTML

Berdasarkan pada gambar 2.4, terdapat beberapa baris yang berisi simbol-simbol. Simbol-simbol ini disebut sebagai tag. Definisi dari tag sendiri yaitu sebuah kode yang biasa digunakan untuk memoles (me-mark-up) teks ASCII menjadi sebuah file dalam bentuk *HTML*. Dalam sebuah *HTML*, setiap teks diapit dengan menggunakan simbol tanda kurung runcing. Sesuai contoh diatas diawali dengan tag pembuka dan penutup seperti <HTML> dan ditandai dengan tanda slash (garis miring) seperti </HTML> didepan penulisan teks. Tag seperti contoh diatas dapat memberikan makna bahwa teks yang ditulis diantara kedua tag tersebut merupakan isi dari dokumen HTML. Website yang dibuat dengan HTML ini, dapat dilihat oleh semua orang yang terkoneksi dengan jaringan internet. Tentunya dengan menggunakan sebuah aplikasi mesin penjelajah internet (browser) seperti Mozilla Firefox, Safari, Internet Explorer dan Google Chrome.

2.7.2 PHP (*Hypertext Processor*)

PHP atau (*Hypertext Preprocessor*) merupakan bahasa *server-side-scripting* yang mendukung *HTML* untuk melakukan pembuatan halaman web yang dinamis (Arief 2011c:43). Sintaks dan perintah-perintah PHP akan langsung dieksekusi di dalam server, karena PHP merupakan *server-side-scripting*. Kemudian hasilnya akan dikirim ke browser dengan format *HTML*. Dengan demikian, user tidak akan dapat melihat kode program yang telah ditulis dalam PHP, sehingga halaman web lebih terjamin keamanannya. PHP dirancang untuk membuat halaman web secara dinamis, yaitu halaman web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke dalam halaman web itu sendiri.

2.7.3 CSS (*Cascading Style Sheet*)

CSS (*Cascading Style Sheet*) memiliki fungsi sebagai pengatur *style* element yang ada pada halaman website. *Cascading Style Sheet* menjadi sebuah rangkaian instruksi yang berguna dalam penentuan bagaimana suatu teks itu dapat terlihat pada halaman web (Wahyu Sya'ban, 2010 :37). Untuk melakukan perancangan teks tersebut, dapat dilakukan dengan mengartikan fonts (huruf), margins (ukuran), colors (warna), latar belakang (background), ukuran font (font sizes) dan lain sebagainya. Contoh elemen-elemen seperti colors (warna), fonts (huruf), sizes (ukuran), serta spacing (jarak) disebut juga dengan *style*. Penggunaan dari *Cascading Style Sheet* yaitu dengan tujuan untuk memperoleh konsistensi *style* pada element tertentu.

2.7.4 JQuery

JQuery merupakan *library* JavaScript untuk menyederhanakan dokumen HTML, membuat *event*, *animasi*, dan *interaksi Ajax* guna mengembangkan web yang dapat diakses lebih cepat. JQuery dibuat untuk mengubah penulisan JavaScript. Dengan JQuery, tampilan website akan

menjadi lebih menarik, interaktif dan *load* akses website yang juga menjadi ringan dan lebih cepat (Aloysius Sigit W, 2011:1).

2.7.5 Bootstrap

Bootstrap yaitu suatu library framework *Cascading Style Sheet* yang dibuat secara khusus untuk digunakan oleh bagian pengembangan di bagian *front-end*. Bootstrap menjadi kerangka kerja *front-end* secara gratis untuk pengembangan *web* yang lebih cepat dan mudah (Husein Alatas, 2013). BOOTSTRAP mencakup template desain yang berbasis *HTML* dan *CSS* untuk *tipografi*, bentuk *tombol*, *tabel*, *navigasi*, *modals*, *korsel gambar*, *plugin JavaScript opsional* dan masih banyak lagi. Bootstrap juga memungkinkan untuk digunakan sebagai pembuatan suatu tampilan *front-end* dengan menambahkan *class-class* tertentu, sehingga dapat lebih mudah dan cepat. Selain itu Bootstrap juga telah menyediakan kumpulan komponen *class interface* dasar yang sudah dirancang untuk menciptakan suatu tampilan yang menarik, bersih, dan ringan.

2.8 Perangkat Lunak yang Digunakan

2.8.1 XAMPP

XAMPP merupakan suatu perangkat lunak bebas yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Nama XAMPP adalah gabungan dari X(sistem operasi) yaitu *Apache*, *PHP*, *Perl*, dan *MySql* yang memberikan interface yang lebih mudah dan praktis (Buana, 2014:4). XAMPP digunakan sebagai *server* yang mampu berdiri sendiri (*localhost*) dan penerjemahannya menggunakan bahasa pemrograman. XAMPP memberi kebebasan untuk dikembangkan lebih lanjut.

2.8.2 Sublime Text

Sublime Text merupakan salah satu sebuah aplikasi editor khusus kode dan teks yang dapat dijalankan pada berbagai platform *operating system* dengan didukung menggunakan teknologi Python API. Aplikasi

tersebut tercipta karena terinspirasi dari terciptanya sebuah aplikasi Vim. Aplikasi tersebut memiliki sifat powerfull dan fleksibel. Selain itu, fungsionalitas dari aplikasi tersebut dapat dikembangkan menggunakan sublime-packages. Sublime Text merupakan beberapa fitur pengembangan fungsionalitas (*packages*) dari aplikasi tersebut, dimana merupakan hasil dari temuan yang mendapat dukungan penuh dari komunitas serta memiliki linsensi aplikasi secara gratis. Sublime Text bukan merupakan sebuah aplikasi *opensource* maupun aplikasi yang dapat digunakan dan didapatkan secara cuma-cuma atau gratis.

Sublime Text dapat menjadi pendukung berbagai macam bahasa pemrograman dan dapat menyajikan fitur *syntax highlight*, dimana mayoritas pada semua bahasa pemrograman didukung oleh komunitas seperti C, C++, C#, CSS, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile dan XML. Umumnya aplikasi ini dapat lebih dimaksimalkan atau didukung dengan menggunakan add-ons yang bisa didownload sesuai dengan kebutuhan *user* bagi bahasa pemrograman yang didukung ataupun belum terdukung secara *default*.

2.9 Pengujian Perangkat Lunak

2.9.1 Black Box Testing (Pengujian Black Box)

Black box testing merupakan suatu pengujian yang dilakukan berdasarkan fungsi dari sebuah program. Tujuan dari *black box testing* yaitu untuk menemukan kesalahan-kesalahan fungsi dari sebuah program yang telah dibuat. *Black box testing* dapat dilakukan dengan pemberian beberapa *input* pada program aplikasi, kemudian dilanjutkan dengan proses yang sesuai dengan kebutuhan fungsionalnya agar dapat melihat apakah program menghasilkan suatu *output* yang diinginkan atau tidak. Serta apakah program tersebut telah sesuai dengan fungsi dari program. Jika *input* yang telah di proses menghasilkan *output* yang sesuai dengan kebutuhan fungsionalnya, maka program yang bersangkutan dapat dinyatakan benar,

namun ketika *output* yang dihasilkan menunjukkan hasil yang belum sesuai dengan kebutuhan fungsionalnya, maka dapat dinyatakan program aplikasi tersebut masih mengandung unsur kesalahan.

Pendapat dari seorang ahli Pressman (2010), mengatakan bahwa *black box testing* juga disebut pengujian suatu tingkah laku, memusat pada kebutuhan fungsional perangkat lunak. Teknik pengujian *black box* memungkinkan memperoleh serangkaian kondisi masukan yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Beberapa jenis kesalahan yang dapat diidentifikasi adalah fungsi tidak benar atau hilang, kesalahan antar muka, kesalahan pada struktur data (pengaksesan basis data), kesalahan performansi, kesalahan inisialisasi dan akhir program.