

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Teori Umum

2.1.1 Perancangan Sistem

Menurut Taylor dalam buku Prahasta (2014: 488) perancangan merupakan proses penggunaan berbagai prinsip dan teknik untuk tujuan pendefinisian perangkat, proses, atau sistem hingga ke tingkat detail tertentu yang memungkinkan realisasi fisiknya.

Perancangan sistem menggambarkan bagaimana suatu sistem dibentuk, yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi. Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem. (Husda, 2012: 137).

2.1.2 Sistem

2.1.2.1 Pengertian Sistem

Menurut Ali dan Wangdra (2010: 8), sistem adalah kumpulan dari sub-sub sistem, elemen-elemen, prosedur-prosedur, yang saling berinteraksi untuk mencapai tujuan tertentu, seperti informasi, target atau *goal*.

Menurut Puspitawati dan Anggadini (2011: 2), sesuatu dapat dikatakan sistem apabila memenuhi dua syarat:

1. Memiliki bagian-bagian yang saling berintegrasi dengan maksud untuk mencapai suatu tujuan, bagian-bagian itu dinamakan subsistem.
2. Harus memenuhi 3 unsur *Input-Process-Output*.

Pengertian sistem menurut pendapat para ahli diantaranya adalah sebagai berikut:

1. Menurut Jerry FithGerald dalam buku Husda (2012: 111), sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
2. Menurut Ludwig Von Bartalanfy dalam buku Husda (2012: 112), sistem merupakan seperangkat unsur yang saling terkait dalam suatu antara relasi diantara unsur-unsur tersebut dengan lingkungan.
3. Menurut L.Ackof dalam buku Husda (2012: 112), sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya.
4. Menurut James Havery dalam Husda (2012: 112), sistem adalah prosedur logis dan rasional untuk merancang suatu rangkaian komponen yang berhubungan satu dengan yang lainnya dengan bermaksud untuk berfungsi sebagai suatu kesatuan dalam usaha mencapai suatu tujuan yang telah ditentukan.

2.1.2.2 Pengertian Subsistem

Suatu sistem dapat terdiri dari bagian-bagian sistem atau subsistem. Masing-masing subsistem dapat terdiri dari subsistem-subsistem lagi atau komponen-komponen pendukung sistem itu sendiri. Konsep sebuah sistem menuntut perancangannya untuk mempertimbangkan sistem sebagai suatu keseluruhan. Keseluruhan sistem terlalu besar untuk dianalisis secara terperinci. Oleh karena itu sistem dibagi atau diuraikan atas beberapa subsistem.

Menurut Norman L. Enger dalam buku Sutabri (2012: 10) subsistem adalah serangkaian kegiatan yang dapat ditentukan identitasnya yang berhubungan dalam suatu sistem.

2.1.2.3 Karakteristik Sistem

Model umum sebuah sistem terdiri dari *input*, *proses* dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengingat sebuah sistem dapat mempunyai beberapa masukan dan keluaran sekaligus. Selain itu sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu.

Menurut Sutabri (2012: 13) karakteristik sistem adalah sebagai berikut:

1. Komponen sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem. Setiap subsistem memiliki sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara

keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan Supra sistem.

2. Batasan sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.

3. Lingkup luar sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkup luar sistem dapat bersifat menguntungkan atau merugikan. Lingkungan luar yang menguntungkan harus tetap dipelihara dan dijaga agar pengaruhnya tidak hilang, sedangkan pengaruh yang merugikan harus dikendalikan. Kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung sistem (*Interface*)

Media penghubung antara subsistem dengan subsistem yang lainnya disebut dengan penghubung sistem atau *interface*. Untuk membentuk suatu kesatuan, sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem yang lainnya. *Output* dari suatu subsistem akan menjadi *input* dari subsistem yang lainnya.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Sebagai contoh di dalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputer. Sementara “data” adalah *signal input* yang akan di olah menjadi informasi.

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti contoh sistem informasi, keluaran yang dihasilkan adalah informasi, dimana informasi ini dapat digunakan sebagai masukan pengambilan keputusan atau hal-hal lain yang merupakan input bagi subsistem lainnya.

7. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran. Sebagai contoh, sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran Sistem (*objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.2.4 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Menurut Sutabri (2012: 15), sistem dapat diklasifikasikan sebagai berikut:

1. Sistem abstrack dan sistem fisik

Sistem abstrak merupakan sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologia yaitu sistem yang berupa pemikiran tentang hubungan manusia dengan Tuhan. Sedangkan Sistem fisik merupakan sistem yang ada secara fisik sehingga setiap makhluk dapat melihatnya. Misalnya sistem komputer, sistem akuntansi, sistem produksi.

2. Sistem alamiah dan sistem buatan manusia

Sistem alamiah merupakan sistem yang terjadi karena melalui proses alam, tanpa campur tangan manusia atau tidak dibuat oleh manusia. Misalnya sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan Sistem buatan manusia adalah sistem yang dirancang oleh manusia. Sistem yang melibatkan interaksi manusia dengan mesin disebut *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

3. Sistem deterministic dan sistem probabilistik

Sistem yang beroperasi dengan tingkahlaku yang sudah dapat diprediksi disebut sitem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer

yang dijalankan. Sedangkan sistem *Probalistik* maksudnya yaitu sistem yang kondisi masa depannya tidak dapat diramalkan atau diprediksi karena mengandung unsur probabilitas.

4. Sistem tertutup dan sistem terbuka

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh terhadap sistem luarnya. Sistem ini bekerja secara otomatis tanpa adanya turut campur tangan dari pihak luarnya. Secara teoritis sistem tersebut ada, tetapi kenyataannya tidak ada sistem yang benar-benar tertutup, yang ada hanya *relatively closed system* (secara relative tertutup, tidak benar-benar tertutup).

Sedangkan Sistem yang terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Lebih spesifik dikenal juga dengan sistem terotomasi, yang merupakan bagian dari sistem buatan manusia dan berinteraksi dengan *control* oleh satu atau lebih komputer sebagai bagian dari sistem yang digunakan dalam masyarakat modern.

2.1.2.5 Daur Hidup Sistem

Menurut Sutabri (2012: 20), siklus hidup sistem (*system life cycle*) adalah proses evolusioner yang diikuti dalam penerapan sistem atau subsistem informasi berbasis komputer. Siklus hidup sistem terdiri dari serangkaian tugas yang mengikuti langkah-langkah pendekatan sistem, karena tugas-tugas tersebut mengikuti pola yang teratur dan dilakukan secara *top down*. Siklus hidup sistem sering disebut sebagai pendekatan air terjun (*waterfall approach*) bagi pembangunan dan pengembangan sistem. Pembangunan sistem hanyalah salah satu dari rangkaian

daur hidup suatu sistem. Meskipun demikian proses ini merupakan aspek yang sangat penting. Beberapa fase/ tahapan daur hidup suatu sistem adalah sebagai berikut:

a. Mengenalinya adanya kebutuhan

Sebelum segala sesuatu terjadi, pastilah terlebih dahulu timbul suatu kebutuhan atau problema yang harus dapat dikenali sebagaimana adanya. Kebutuhan dapat terjadi sebagai hasil perkembangan organisasi. Semua kebutuhan harus didefinisikan dengan jelas. Tanpa adanya kejelasan akan kebutuhan yang ada, pembangunan sistem akan kehilangan arah dan efektivitasnya.

b. Pembangunan Sistem

Suatu proses atau seperangkat prosedur yang harus diikuti guna menganalisis kebutuhan yang timbul dan membangun suatu sistem untuk memenuhi kebutuhan tersebut.

c. Pemasangan Sistem

Setelah tahap pembangunan sistem selesai, kemudian sistem akan dioperasikan. Pemasangan sistem merupakan tahap yang penting dalam daur hidup sistem, dimana peralihan dari tahap pembangunan menuju tahap operasional adalah pemasangan sistem, yang merupakan langkah akhir dalam suatu pembangunan sistem.

d. pengoperasian Sistem

Program-program komputer dan prosedur-prosedur pengoperasian yang membentuk suatu sistem informasi semuanya bersifat statis, sedangkan organisasi yang ditunjang oleh suatu sistem informasi selalu mengalami perubahan karena pertumbuhan kegiatan, perubahan peraturan atau kebijaksanaan, ataupun kemajuan teknologi. Untuk mengatasi perubahan-perubahan tersebut, sistem harus diperbaiki atau diperbaharui.

e. Sistem menjadi usang

Kadang-kadang perubahan yang terjadi secara drastis sehingga tidak dapat diatasi hanya dengan melakukan perbaikan pada sistem yang sedang berjalan. Tiba saatnya dimana secara ekonomis dan teknis, sistem yang sudah ada tidak layak lagi dioperasikan dan sistem yang baru perlu dibangun untuk menggantikannya.

2.1.3. Konsep Dasar Informasi

Menurut Ali dan Wangdra (2010: 8), informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi sipenerima dan mempunyai nilai yang nyata atau dapat dirasakan manfaatnya dalam keputusan-keputusan yang akan datang.

Menurut Puspitawati dan Anggadini (2011: 13), informasi adalah data yang diolah menjadi bentuk yang lebih bergunadan lebih berarti bagi yang menerimanya.

Menurut Sutabri (2012: 30) informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Nilai dari informasi ditentukan dari 2 (dua) hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai apabila manfaat yang diperoleh lebih berharga dibandingkan dengan biaya untuk mendapatkannya.

Nilai informasi didasarkan atas 10 (sepuluh) sifat, yaitu:

1. Mudah diperoleh

Sifat ini menunjukkan kemudahan dan kecepatan untuk memperoleh informasi.

2. Luas dan lengkap

Sifat ini menunjukkan kelengkapan isi informasi. Hal ini tidak hanya mengenai volumenya, akan tetapi juga mengenai keluaran informasinya.

3. Ketelitian

Sifat ini berhubungan dengan tingkat kebebasan dari kesalahan keluaran informasi.

4. Kecocokan

Sifat ini menunjukkan seberapa baik keluaran informasi dalam hubungannya dengan permintaan para pemakai. Isi informasi harus ada hubungannya dengan masalah yang sedang dihadapi sedangkan semua keluaran yang lainnya tidak berguna.

5. Ketepatan waktu

Sifat ini berhubungan dengan waktu yang dilalui, yang lebih pendek dari siklus untuk mendapatkan informasinya.

6. Kejelasan

Sifat ini menunjukkan tingkat kejelasan informasi. Informasi hendaknya terbebas dari istilah-istilah yang tidak jelas.

7. Keluwesan

Sifat ini berhubungan dengan apakah informasi tersebut dapat digunakan untuk membuat lebih dari satu keputusan, tetapi juga apakah dapat digunakan untuk lebih dari seorang pengambil keputusan.

8. Dapat dibuktikan

Sifat ini menunjukkan sejauh mana informasi dapat diuji oleh beberapa pemakai hingga sampai didapatkan kesimpulan yang sama.

9. Tidak ada prasangka

Sifat ini berhubungan dengan ada tidaknya keinginan untuk mengubah informasi tersebut guna mendapatkan kesimpulan yang telah diarahkan sebelumnya.

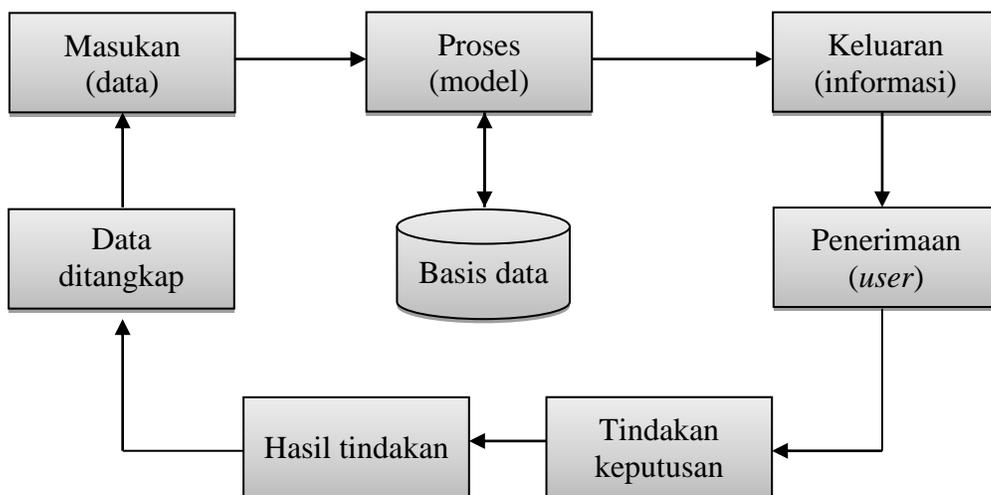
10. Dapat diukur

Sifat ini menunjukkan hakikat informasi yang dihasilkan oleh sistem informasi formal.

Menurut Davis dalam buku Husda (2012: 117) informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima mempunyai nilai yang nyata yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang.

Menurut Mcleod dalam buku Husda (2012: 117) informasi adalah data yang telah dioalah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambilan keputusan saat ini dan mendatang.

Secara umum informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi si penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Data adalah kenyataa yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Data diolah melalui suatu metode menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi ini dapat digambarkan sebagai berikut:



Sumber : Husda (2012: 118)

Gambar 2.1 Siklus informasi

2.1.4. Sistem Informasi

Menurut Sutabri (2012: 38), sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu.

Pada sistem informasi ada beberapa hal yang perlu diperhatikan antara lain:

1. Keadaan (*Reality*)

Yaitu menunjukkan seberapa besar sistem dapat diandalkan untuk melakukan suatu proses yang dapat dipercaya dan dibutuhkan.

2. Ketersediaan (*Availability*)

Yaitu bahwa sistem dapat menyediakan informasi yang dibutuhkan kapanpun oleh pemakai sistem.

3. Keluwesan (*Flexibility*)

Yaitu menunjukkan bahwa sistem mudah beradaptasi sesuai dengan kebutuhan-kebutuhan pemakai yang selalu berubah-ubah.

4. Skedul Instalasi

Yaitu terdiri dari periode waktu antara saat organisasi sadar untuk membutuhkan sistem informasi dan saat sistem tersebut diterapkan.

5. Kemudahan dipelihara

Yaitu setelah sistem diterapkan maka sistem harus dipelihara.

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya.

Sistem informasi dapat didefinisikan sebagai suatu sistem didalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan.

Sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basisdata, dan blok kendali. keenam blok tersebut saling berinteraksi satu dengan yang lainnya membentuk suatu kesatuan untuk mencapai sasaran. Komponen-komponen sistem informasi ini dapat diuraikan sebagai berikut:

1. Blok masukan (*Input Block*)

Input mewakili data yang masuk kedalam sistem informasi. Yang dimaksud dengan input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model (*Model Block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*Technology Block*)

Teknologi merupakan kotak alat (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran dan membantu pengendalian dari sistem secara menyeluruh. Teknologi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data (*Database Block*)

Merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basisdata untuk keperluan penyediaan informasi lebih lanjut.

6. Blok Kendali (*control Block*)

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecuranga, kegagalan pada sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dicegah dan bila terlanjur terjadi maka kesalahan-kesalahan dapat dengan cepat diatasi.

2.1.5. Pengembangan dan Perancangan Sistem Informasi

Pengembangan sistem informasi merupakan tindakan mengubah, mengganti atau menyusun sistem lama menjadi sistem baru baik secara sebagian maupun keseluruhan untuk memperbaiki sistem yang selama ini berjalan (yang telah ada).

2.1.5.1. SDLC (Software Development Life Cycle)

SDLC atau Software Development Life Cycle sering disebut juga System Development adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan untuk mengembangkan sistem-sistem perangkat lunak sebelumnya. Menurut Rosa dan Shalahuddin (2011: 24-26) tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut:

1. Inisiasi (*Initiation*)

Tahapan ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*System Concept Development*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis konsep biaya, manajemen rencana dan pembelajaran kemudahan sistem.

3. Perencanaan (*Planning*)

Mengembangkan rencana pengembangan manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resources*) yang dibutuhkan untuk memperoleh solusi.

4. Analisis kebutuhan (*requirements analysis*).

Menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.

5. Desain (*design*)

Mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem focus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan. Membuat basis data dan mempersiapkan prosedur kasus pengujian, mempersiapkan berkas atau *file* pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program, peninjauan pengujian.

7. Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user*. Menghasilkan laporan analisis pengujian.

8. Implementasi (*implementation*)

Termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operations and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*)

Mendeskripsikan aktifitas akhir dari mengembangkan sistem dan membangun data yang sebenarnya sesuai dengan aktifitas *user*.

Ada beberapa model SDLC yang dapat digunakan. Semuanya memiliki kelemahan dan kelebihan pada setiap model SDLC. Hal yang terpenting adalah mengenali tipe pelanggan (*customer*) dan memilih menggunakan model SDLC yang sesuai dengan karakter pelanggan (*customer*) dan sesuai dengan karakter pengembang. Menurut Rosa dan Shalahuddin (2011: 26-37) model-model SDLC adalah sebagai berikut:

1. Model *waterfall*

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linear (*sequential linier*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

2. Model *prototype*

Model *protptype* dimulai dari pengumpulan kebutuhan pelanggan terhadap perangkat lunak yang akan dibuat. Lalu dibuatlah program *protptype* agar pelanggan lebih terbayang dengan apa yang sebenarnya diinginkan. Program *protptype* biasanya merupakan program *protptype* yang belum jadi.

3. Model *rapid application Development* (RAD)

Model *rapid application Development* (RAD) adalah model proses pengembangan perangkat lunak yang bersifat incremental terutama untuk waktu pengerjaan yang pendek. Model RAD adalah adaptasi dari model air terjun untuk mengembangkan setiap komponen perangkat lunak.

4. Model *iterative*

Model interatif mengkombinasikan proses-proses pada model air terjun dan intertif pada *prototype*. Model *incremental* akan menghasilkan versi-versi perangkat lunak yang sudah mengalami penambahan fungsi untuk setiap pertambahannya (*inkremen/increment*).

5. Model *spiral*

Model *spiral* memasangkan *interatif* dan model *Prototype* dengan *control* dan aspek sistematis yang diambil dari model air terjun.

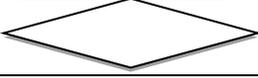
Pada penyusunan penelitian ini, diantara model SDLC diatas, peneliti menggunakan pendekatan model *waterfall* sebagai perancangan sistem informasi persediaan berbasis *web*.

2.1.5.2. *Flowchart*

Menurut Puspitawati dan Anggadini (2011: 105), *flowchart* dapat dibagi menjadi tiga bagian yaitu *flowmap* atau document *flowchart*, sistem *flowchart* dan program *flowchart*.

Flowmap merupakan diagram alir yang menunjukkan arus dari dokumen, aliran data fisik, entitas, entitas sistem informasi dan kegiatan operasi yang digunakan dengan sistem informasi. Sistem *flowchart* atau diagram alir sistem adalah diagram yang menggambarkan struktur program atau deskripsi program untuk setiap modul program suatu sistem berbasis komputer. Program *flowchart* merupakan diagram yang menggambarkan urutan langkah detail dan logika program. Dalam suatu informasi, program *flowchart* merupakan diagram secara detail dari proses-proses didalam sistem *flowchart*. Berikut simbol atau notasi baku yang digunakan adalah sebagai berikut:

Tabel 2.1 Simbol *Flowchart*

Simbol	Deskripsi
	Dokumen I/O
	Manual <i>Input</i>
	<i>Terminator</i>
	Proses Manual
	Proses
	Pemilihan
	Tape
Simbol	Deskripsi
	<i>Display</i>
	Penghubung Berbeda Halaman
	<i>File</i>
	<i>Predefined Process</i>
	Penghubung Satu Halaman
	<i>Database</i>
	Arsip
	Arah Proses

Sumber: Puspitawati dan Anggadini (2011: 105)

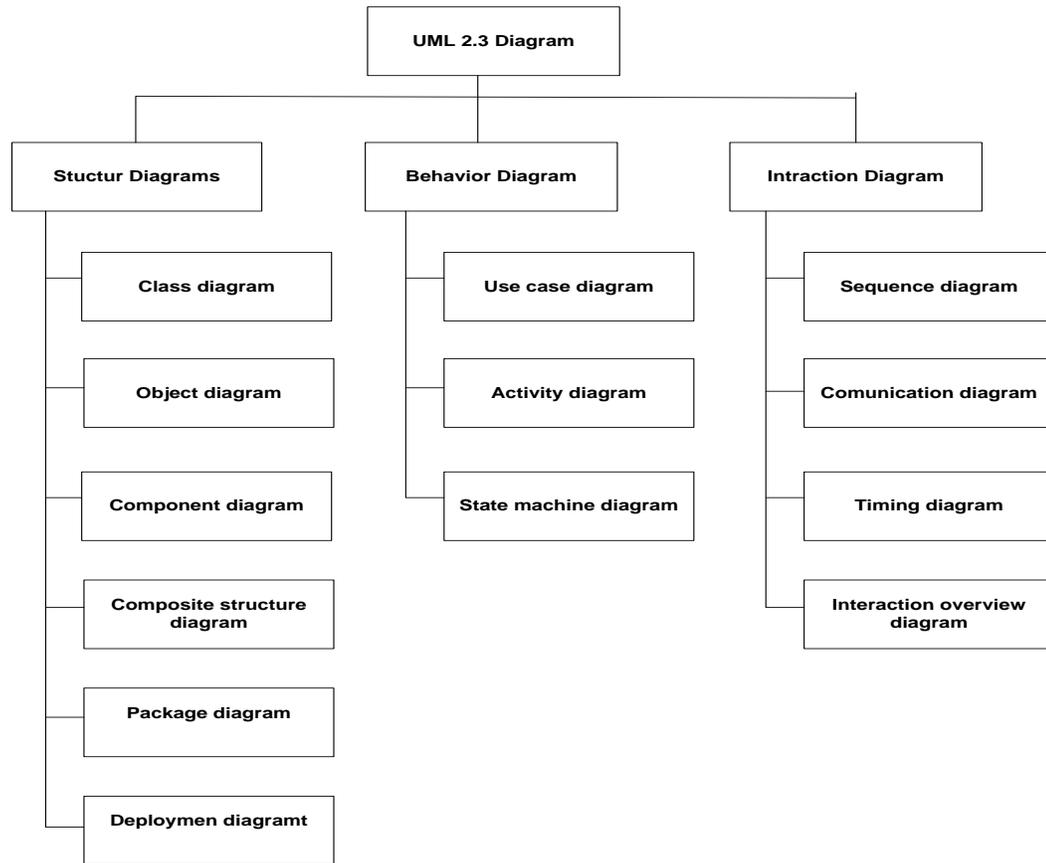
2.1.5.3. UML (*Unified Modeling Language*)

Perkembangan teknik pemrograman berorientasi objek, munculah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

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.

2.1.5.3.1. Diagram UML

Diagram UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini:



Sumber: Rosa dan Shalahuddin (2011: 121)

Gambar 2.2 Diagram UML

Berikut ini penjelasan singkat dari pembagian kategori tersebut:

1. *Structure diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagram* yaitu diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antara subsistem pada suatu sistem.

Dari 13 diagram dalam melakukan perancangan sistem informasi persediaan penulis menggunakan *Class diagram*, *Use Case diagram*, *Activity diagram*, dan *Sequence Diagram* berikut penjelasan dari masing-masing diagram:

1. *Class Diagram*

Diagram kelas atau class diagram menggambarkan struktur sistem dari pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki atribut dan metode atau operasi. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Dalam mendefinisikan metode yang ada di dalam kelas perlu memperhatikan *cohesion* dan *coupling*. *Cohesion* adalah ukuran seberapa dekat keterkaitan instruksi di dalam sebuah metode terkait satu sama lain sedangkan *coupling* adalah ukuran seberapa dekat keterkaitan instruksi antara metode yang satu dengan metode yang lain dalam suatu sebuah kelas. Dalam diagram kelas terdapat beberapa symbol dalam penggunaannya (Rosa dan Shalahuddin, 2011:122).

Tabel 2.2 Simbol Class Diagram

Simbol	Deskripsi
--------	-----------

<p style="text-align: center;">Kelas</p> <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> <p style="text-align: center;">nama_kelas</p> <p style="text-align: center;">+atribut</p> <p style="text-align: center;">+operasi()</p> </div>	Kelas pada struktur system
<p style="text-align: center;">Antarmuka / <i>Interface</i></p> <div style="text-align: center;">  <p>nama_interface</p> </div>	Sama dengan konsep <i>interface</i> dalam pemograman berorientasi objek
<p style="text-align: center;">Asosiasi / <i>association</i></p> <p style="text-align: center;">—————</p>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p style="text-align: center;">Asosiasi berarah / <i>directed association</i></p> <p style="text-align: center;">—————></p>	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
<p style="text-align: center;">Generalisasi</p> <p style="text-align: center;">—————▶</p>	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
<p style="text-align: center;">Kebergantungan / <i>dependency</i></p> <p style="text-align: center;">.....></p>	Relasi antara kelas dengan makna kebergantungan antar kelas
<p style="text-align: center;">Agregasi / <i>aggregation</i></p> <p style="text-align: center;">—————◊</p>	Relasi antar kelas dengan makna semua bagian (<i>whol -part</i>)

Sumber: Rosa dan Shalahuddin (2011: 123)

2. Use Case Diagram

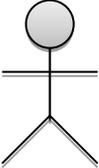
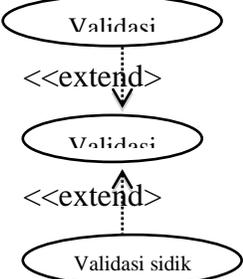
Use case diagram merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

- a. Aktor merupakan orang, proses, atau sistem lain berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

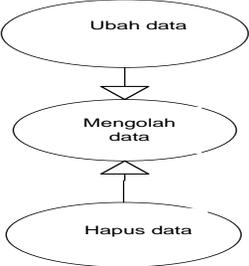
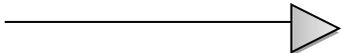
b. *Use case*: merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor.

Simbol dalam *Use case Diagram*:

Tabel 2.3 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p data-bbox="527 663 636 695"><i>Use case</i></p> 	<p>Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal di awal frase nama <i>use case</i>.</p>
<p data-bbox="527 898 685 930">Aktor / <i>actor</i></p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.</p>
<p data-bbox="407 1207 667 1239">Asosiasi / <i>association</i></p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan yaitu , mirip dengan prinsip <i>inheritance</i> pada pemrograman objek, <i>use</i> berorientasi biasanya</p> 

Lanjutan table 2.3

	<p><i>case</i> tambahan memiliki nama depan sama dengan <i>use case</i> yang ditambahkan misal Arah panah mengarah pada <i>use case</i> yang ditambahkan</p>
<p>generalisasi / <i>generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:</p> <p>Arah panah mengarah pada <i>use case</i> yang menjadi generalisasi (umum)</p> 
<p>Menggunakan / include / <i>uses</i></p> <p><<include>></p>  <p><<uses>></p> 	<p>Relasi <i>use case</i> tambah ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i>. Ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>usecase</i>:</p> <p><i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> yang ditambahkan dijalankan.</p> <p><i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan. Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan</p>

Sumber: Rosa dan Shalahuddin (2011: 131)

3. *Activity Diagram*

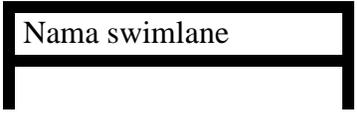
Activity Diagram (diagram aktivitas) menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah system atau proses bisnis. *Activity Diagram* juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- a. Rancangan proses bisnis di mana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b. Urutan atau pengelompokan tampilan dari sistem/ *userinterface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- c. Rancangan pengujian di mana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Menurut Rosa dan Shalahuddin (2011:135) *Diagram Activity* memiliki beberapa simbol dalam penggunaannya. Berikut adalah simbol-simbol yang ada pada *Diagram Activity*:

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Deskripsi
--------	-----------

Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / <i>joint</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status Akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

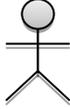
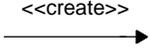
Sumber: Rosa dan Shalahuddin (2011: 134)

4. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dalam mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Banyaknya diagram sekuen yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen.

Menurut Rosa dan Shalahuddin (2011:138) *Diagram Sequence* memiliki beberapa simbol dalam penggunaannya. Berikut adalah simbol-simbol yang ada pada *Diagram Sequence*:

Tabel 2.5 Simbol *Sequence Diagram*

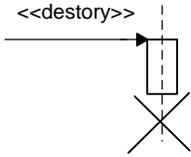
Simbol	Deskripsi
actor atau   tanpa waktu aktif	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem yang akan dibuat itu sendiri, jadi walaupun simbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor
Garis hidup / <i>lifeline</i> 	Menyatakan kehidupan suatu objek
Objek 	Menyatakan objek yang berinteraksi pesan
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan
Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
Pesan tipe call	Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri, arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil

1: nama_metode()



operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi

Lanjutan tabel 2.5

<p>Pesan tipe <i>send</i></p> <p>1:Masukan </p>	<p>Menyatakan bahwa suatu objek mengirimkan data /masukan /informasi ke objek yang lainnya, arah panah mengarah pada objek yang dikirim</p>
<p>Pesan tipe <i>return</i></p> <p>1: Keluaran </p>	<p>Menyatakan bahwa suatu objek telah menjalankan suatu operasi / metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> <p></p>	<p>Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i></p>

Sumber: Rosa dan Shalahuddin (2011: 138)

2.2. Tinjauan Teori Khusus

2.2.1 Persediaan

Menurut Raymond McLeod (1998) dalam Mujianti dan sukandi (2016: 12) persediaan adalah sebagai suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam periode usaha yang normal.

Dalam melakukan aktivitas, setiap perusahaan baik perusahaan jasa maupun perusahaan manufaktur pasti akan memerlukan adanya persediaan atau *inventory*. Tanpa persediaan, perusahaan akan dihadapkan pada resiko besar yaitu tidak terpenuhinya permintaan produk pada waktu yang diinginkan, tetapi sebaliknya jika perusahaan memiliki persediaan yang berlebihan maka akan menimbulkan adanya biaya yang disebut dengan biaya penyimpanan.

2.2.2. Obat

Obat adalah bahan atau panduan yang dimaksudkan untuk mendapatkan diagnose, mencegah, menghilangkan, menyembuhkan gejala penyakit, kelainan badaniah atau rohaniyah pada manusia atau hewan untuk memperelok bahan atau bagian tubuh manusia (Mujianti dan Sukandi, 2016: 12)

2.2.3. Apotek

Menurut PP No.51 Tahun 2009 dalam Nurdiansyah dan Triyono (2013: 47) Apotek adalah sarana pelayanan kefarmasian tempat dilakukan praktik kefarmasian oleh apoteker. Pelayanan kefarmasian adalah suatu pelayanan langsung dan bertanggung jawab kepada pasien yang berkaitan dengan sediaan farmasi dengan maksud mencapai hasil yang pasti untuk meningkatkan mutu kehidupan pasien. Pekerjaan kefarmasian yang dilakukan meliputi pembuatan termasuk pengendalian mutu sediaan farmasi, pengamanan, pengadaan, penyimpanan, dan pendistribusian atau penyaluran obat, pengelolaan obat, pelayanan obat atas resep dokter, pelayanan informasi obat, serta pengembangan obat, bahan obat, obat tradisional dan kosmetika.

2.2.4. WEB

Menurut Sidik dan Pohan (2012: 1), *Word Wide Web(WWW)*, lebih dikenal dengan *web*, merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. *Web* pada awalnya adalah ruang informasi dalam internet, dengan menggunakan teknologi *hyperteks*, pemakai dituntun untuk menemukan

informasi dengan mengikuti link yang disediakan dalam dokumen *web* yang ditampilkan dalam browser *web*. *Web* memudahkan pengguna komputer untuk berinteraksi dengan pelaku internet lainnya dan menelusuri informasi di internet. Selain itu *web* telah diadopsi oleh perusahaan sebagai sebagian dari strategi teknologi informasinya, karena beberapa alasan:

1. Akses informasi mudah
2. *Setup server* lebih mudah
3. Informasi mudah distribusikan
4. Bebas *platform*; informasi dapat disajikan oleh *browser web* pada sistem operasi mana saja karena standar dokumen berbagai tipe data dapat disajikan.

2.2.5. HTML5

HTML5 layaknya sebuah *HTML* biasa yang sering digunakan dalam membangun aplikasi *web*, *HTML5* memiliki keunggulan dibanding versi terdahulunya. *HTML5* mampu menyederhanakan kode-kode *HTML* terdahulu menjadi lebih ringkas. *HTML5* merupakan revisi ke-5 dari *HTML* dan saat ini juga masih dalam tahap pengembangan. *HTML5* konon akan menjadi standar pemrograman *web* menggantikn *HTML* versi terdahulu. Berikut adalah fitur-fitur terbaru dalam *HTML 5* menurut Agus Saputra (2012: 17) sebagai berikut:

1. Unsur canvas untuk gambar.
2. Bentuk *control form* seperti kalender, waktu, *email*, *url*, dan *search*.
3. Elemen konten yang lebih spesifik, seperti *artikel*, *footer*, *header*, *navigasi*, dan *section*.

4. Dukungan yang lebih baik untuk penyimpanan secara *offline*.
5. Adanya dukungan untuk pemutaran *video* dan *audio*.

Berikut adalah elemen-elemen baru yang ada pada *HTML5*:

1. *Section*.
2. *Article*, bias berupa entri blog atau tulisan konten.
3. *Aside*, menyajikan konten pelengkap.
4. *Header*, digunakan untuk judul, deskripsi, bahkan nav atau *navigasi*.
5. *Footer*, digunakan untuk bagian bawah *web* yang digunakan untuk menerangkan informasi *copyright* (hak cipta), perusahaan, nama pemuat, kontak, dan sebagainya.
6. *Dialog*, yang dikombinasikan dengan *dt* dan *dd* digunakan untuk menyajikan percakapan.
7. Penggunaan elemen *figure*, *video*, *audio*, *source*, *embedded*, *canvas*, dan elemen-elemen lainnya yang berkaitan dengan *multimedia*.

2.2.6. CSS3

CSS merupakan singkatan dari *Cascading Style Sheet* merupakan bahasa pemrograman *web* yang didesain khusus untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* lebih rapi terstruktur, dan seragam. Tujuan utama *CSS* adalah untuk memisahkan konten utama dengan tampilan dokumen lainnya. *Web* yang menggunakan *CSS* akan lebih ringan dan

mudah untuk dibuka dibandingkan dengan *web* yang tidak menggunakan *CSS*. (Agus Saputra, 2012: 29).

CSS saat ini sudah mencapai versi 3 dimana pada setiap versi pasti ada peningkatan yang dilakukan. Menurut Agus Saputra (2012: 28) terdapat peningkatan tiap versi *CSS* dari versi 1 sampai versi 3 yaitu:

1. *CSS1*, masih kuno, *CSS* hanya dikembangkan dan digunakan untuk formatting dokumen html.
2. *CSS2*, disini sudah mulai menggunakan *font*, *table-layout*, dan berbagai media printer.
3. *CSS3*, merupakan pengembangan dari versi *CSS* sebelumnya. Peningkatan yang mencolok pada versi ini adalah peningkatan fitur yang mengarah pada efek animasi.

2.2.7. PHP

PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. *PHP* merupakan bahasa pemrograman skrip yang diletakan dalam server yang biasa digunakan untuk membuat aplikasi *web* yang bersifat dinamis. Maksud *web* dinamis adalah dapat membentuk suatu tampilan *web* berdasarkan permintaan terkini, dapat dilakukan dengan menampilkan isi *database* ke halaman *web*. *PHP* juga digunakan secara *command line*, yaitu skrip *PHP* dapat dijalankan tanpa melibatkan *web server* maupun *browser* (Nurdiansyah dan Triyono, 2013: 47).

Menurut Aditya (2011: 2) beberapa kelebihan PHP dari bahasa pemrograman *web*, antara lain:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache*, *IIS*, *lightpd*, hingga *Xitami* dengan *konfigurasi* yang *relative* mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahaman, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara *runtime* melalui *console* serta dapat juga menjalankan perintah-perintah *system*.

2.2.8. MySQL

MySQL merupakan salah satu database kelas dunia yang cocok dipadukan dengan bahasa pemrograman *PHP*. *MySQL* bekerja menggunakan bahasa *SQL* (*Structure Query Language*) yang merupakan bahasa standar yang digunakan untuk manipulasi database.

Perintah yang paling sering digunakan dalam *MySQL* adalah *SELECT* (mengambil), *INSERT* (menambah), *UPDATE* (mengubah), dan *DELETE* (menghapus). *SQL* juga menyediakan perintah untuk membuat *database*, *field*, ataupun *index* untuk menambah atau menghapus data.

Menurut Agus Saputra (2012: 78) ada beberapa alasan yang menjadikan database *MySQL* sangat diminati oleh para programmer, diantaranya:

1. Bersifat *open source*.
2. Menggunakan bahasa *SQL*, yang merupakan standar bahasa dalam pengolahan data.
3. *Performance* dan *reliable*, pemrosesan *database*-nya sangat cepat dan stabil.
4. Sangat mudah dipelajari (*ease of use*).
5. Memiliki dukungan (*group*) pengguna *MySQL*.
6. Lintas *Platform*, dapat digunakan pada berbagai Sistem Operasi berbeda.
7. *Multiuser*, dimana *MySQL* dapat digunakan oleh banyak *user* dalam waktu yang bersamaan tanpa mengalami konflik.

2.2.9. XAMPP

Menurut Aditya (2011: 3), *XAMPP* adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program.

Fungsi dari *XAMPP* adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP server*, *My SQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam GNU yaitu sigkatan dari *General Public*

License dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis.

2.2.10. Adobe Dreamweaver CS6

Adobe Dreamweaver CS6 merupakan versi terbaru dari adobe dreamweaver yang sebelumnya adalah *adobe dreamweaver CS5*. Aplikasi *Dreamweaver CS6* memberikan tampilan yang lebih baik dan tentu saja semakin mudah dalam penggunaannya. Aplikasi ini mengintegrasikan beragam fitur untuk memenuhi kebutuhan pengembangan *website*, termasuk pembuatan halaman *web* dan pengelolaannya.

Adobe Dreamweaver CS6 menyertakan banyak tool yang berkaitan dengan pengkodean seperti *HTML*, *CSS*, *XML*, dan pemrograman *Client Side*, yaitu *JavaScript* dengan penggunaan yang sangat mudah dan *user friendly*. Aplikasi ini juga mendukung pemrograman *Script Server Sides* seperti *PHP*, *Active ServerPage (ASP)*, *ASP.NET*, *ASP Java Script*, *ASP VB Script*, *Cold Fusion*, dan *Java Server Page (JSP)*.

Fasilitas yang ada dalam *Adobe Dreamweaver CS6* memberikan kemudahan kepada *user* untuk melakukan pengeditan karena ditampilkan secara visual. Penambahan desain dan fungsi pada halaman *web* tidak harus dituliskan dalam baris kode. Anda tinggal memilih dan menempatkan komponen *web* dengan melakukan drag kedalam dokumen *web* secara langsung dan cepat. *Adobe Dreamweaver CS6* juga dapat meng-import dan menyisipkan *image* atau *movie* yang dibuat dari aplikasi lainnya seperti *file flash (.SWF)* ataupun *FLV*.

2.2.10.1 Fitur Baru *Adobe Dreamweaver CS6*

Adobe Dreamweaver CS6 memiliki fitur-fitur baru yang merupakan penyempurnaan dari versi sebelumnya. Fitur-fitur baru yang ditambahkan sebagai berikut:

1. *Buil-in CMS*

Dukungan untuk menciptakan dan pengujian bahan-bahan untuk sistem manajemen konten seperti *WordPress, Joomla!, dan Drupal*.

2. *Integrasi dengan Adobe Browser Lab*

Pemetaan halaman dinamis dan konten local dengan melihat beberapa pandangan, diagnosis, dan perbandingan.

3. Petunjuk *PHP* kelas custom

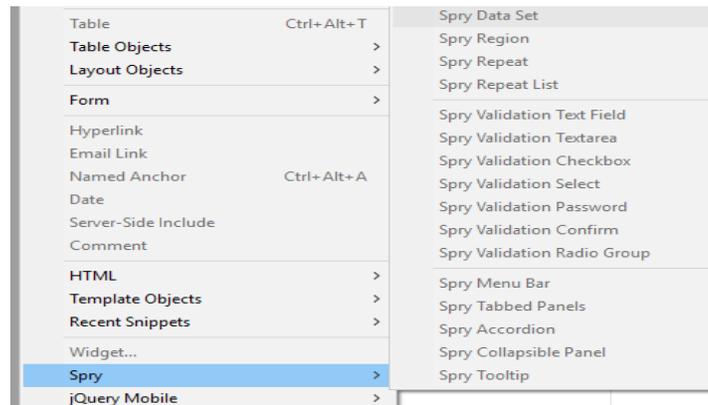
Tampilan sintaks yang tepat untuk fungsi *PHP* dimaksudkan mencegah kesalahan dalam pengkodean.

4. Manajemen *file* yang digunakan

Mengatur berbagai macam *file* yang digunakan untuk menyusun halaman *web* dengan lebih efisien. Fitur ini juga menampilkan semua dokumen yang dihubungkan pada halaman *web* seperti *CSS, JavaScript, PHP*, atau *XML* dalam satu baris disebelah atas pada dokumen *window*.

5. *HTML data sets*

Dengan fitur ini, anda dapat membuat data dalam tabel *HTML, div tags*, atau *unordered list*. Caranya adalah pada menu bar pilih *Insert → Spry → Spry Data Set*.



Gambar 2.3 Pilihan *Spry Data Set*

6. *Photoshop Smart Objects*

Fitur ini memungkinkan anda dapat melakukan *copy paste file Photoshop* (*.PSD) ke dalam halaman *web* pada aplikasi *Adobe Dreamweaver CS6* dan langsung mengedit *file* tersebut sebagai desain *interface web*.

2.3 Penelitian Terdahulu

Penelitian terdahulu yang digunakan sebagai bahan penimbang dalam penelitian ini. Sesuai dengan judul maka penelitian terdahulu yang berkaitan adalah sebagai berikut:

Tabel 2.6 Penelitian Terdahulu

No.	Penelitian dan tahun	Judul	Hasil
1	Hanik Mujianti dan Sukandi, Volume 4 No.1 Maret 2016 ISSN: 2338-8145	Analisa dan Perancangan sistem Informasi Stock Obat Pada Apotek Arjowinangun	Sistem stok obat Apotek pada Arjowinangun saat ini adalah sistem konvensional yaitu dengan cara pembukuan, yang dapat menimbulkan masalah diantaranya

			ketidak tersediaan obat, keterlambatan obat hal ini akan relative memakan waktu cukup lama. Tujuan penelitian ini untuk menghasilkan perancangan sistem informasi stok obat pada Apotek Arjowinangun.
2.	Titin Sri Handayani dan Indah Uly Wardati, Volume 1 No. 1 2014 ISSN: 2355-1313	Analisis dan Perancangan Sistem Informasi Pengolahan Data Persediaan Obat pada Pos Kesehatan Desa Wonoanti	Sistem informasi yang manual akan sangat lambat dan membutuhkan waktu yang cukup lama untuk dapat menyelesaikannya. Berbeda dengan pengolahan data yang terkomputerisasi, cara kerjanya akan cepat, efektif dan efisien. Sehingga membantu orang-orang yang mengerjakan tugas tersebut.

Lanjutan table 2.6

3.	Deni Eko Purwanto, Volume 5 No. 3 2013 ISSN: 1979-9330	Pembangunan Sistem informasi Apotek Pink Pacitan	<p>Apotek Pink Pacitan merupakan salah satu Apotek yang mengatur manajemen secara konvensional, dan belum memanfaatkan teknologi komputer secara optimal. Pembuatan sistem informasi yang dapat digunakan untuk pendataan obat</p> <p>pada Apotek memberikan kemudahan dalam proses pengolahan data Apotek, seperti data obat, dan penjualan obat, pembuatan laporan kepada pimpinan dan dapat mengetahui obat yang mendekati kadaluarsa.</p>
4.	Nurdiansyah dan Ramadian Agus Triyono, Volume 2 No. 3 2013 ISSN: 2302-5700	Pembuatan Sistem Informasi Berbasis <i>Web</i> pada Apotek Tulakan	<p>Dengan adanya sistem informasi Apotek, maka akan mempermudah proses pencarian dan mengupdate suatu data untuk sebuah informasi yang lebih akurat dan lebih tepat waktu sesuai dengan yang diharapkan dan yang diinginkan pemakai atau user. Hasil penelitian ini mempermudah dalam proses pengolahan data Apotek, seperti data obat, data penjualan, pembuatan kuitansi, pembuatan laporan kepada pimpinan dan untuk mempermudah dalam proses pencarian data selain itu memiliki media penyimpanan yang lebih efektif dan lebih besar.</p>
5.	Triyono, Volume 3 No. 4 2014 ISSN: 2302-5700	Pembuatan Sistem Pencatatan stok Obat pada UPT Puskesmas Kebonagung	<p>Teknologi informasi banyak diaplikasikan sebagai <i>website</i> untuk menyebarkan informasi secara <i>online, web</i> profil puskesmas untuk menyebarkan tentang kesehatan dan disesuaikan dengan kebutuhan puskesmas untuk masyarakat luas.</p>