BAB II
TINJAUAN PUSTAKA

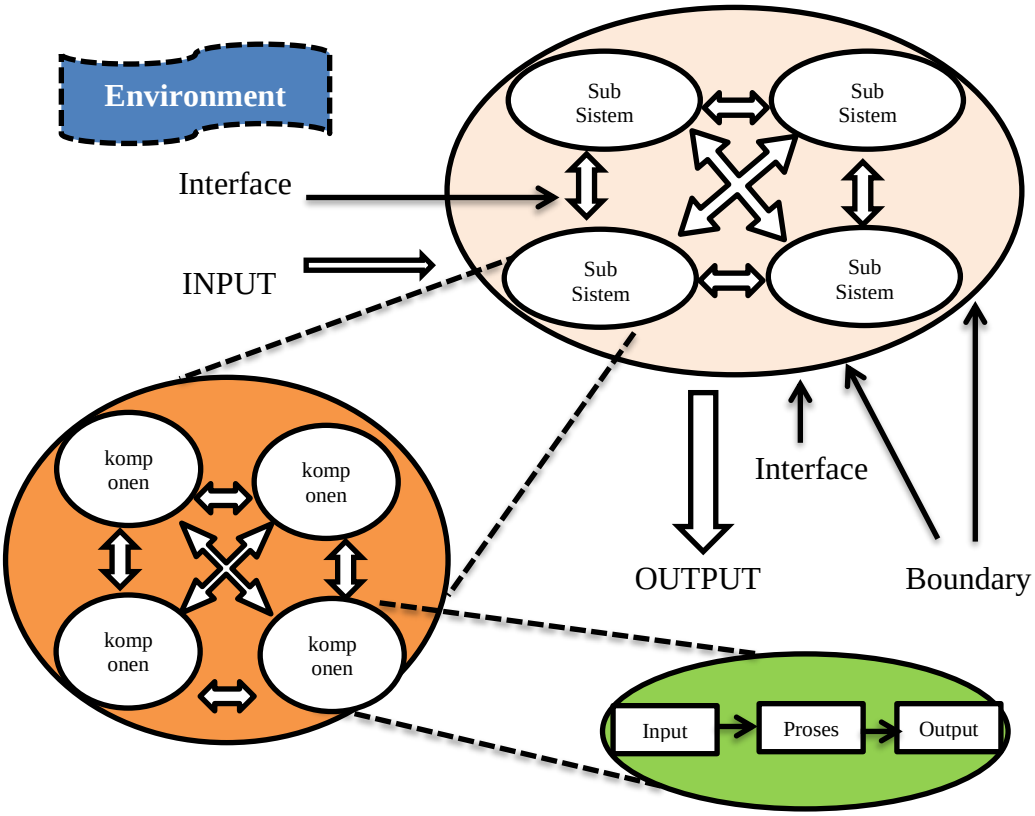
2.1 Tinjauan Teori Umum

Dalam penulisan skripsi ini, penulis menggunakan teori umum dan teori khusus yang menjadi patokan dalam penulisan skripsi ini dan sebagai bahan acuan. Teori adalah seperangkat konsep, defenisi dan proposisi yang tersusun secara sistematis sehingga dapat digunakan untuk menjelaskan dan meramalkan fenomena (Sugiyono, 2012 : 52).

2.1.1 Sistem

Sistem (*system*) adalah kumpulan dari sub-sub sistem, elemen-elemen, prosedur-prosedur, yang saling berintegrasi untuk mencapai tujuan tertentu, seperti informasi, target atau goal. Karakter suatu sistem terdiri dari : komponen (*Components*), Batas Sistem (*Boundary*), Lingkungan luar Sistem (*Environments*), Penghubung (*Interface*), *input*, *process* dan *output*, Sasaran (*Objectives*), Tujuan (*Goal*). Menurut buku *Conceptual, Structure and Development*, "Sistem dapat bersifat abstrak atau fisis. Sistem yang abstrak adalah susunan yang teratur dari gagasan-gagasan atau konsep yang saling tergantung" (Ali & Wangdra, 2010 : 8).

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu (Ermatita, 2016).



Gambar 2. 1 Karakteristik Sistem

2.1.2 Karakteristik Sistem

Menurut (Muslihudin & Oktafianto, 2016 : 4-5) sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut dapat dikatakan sebagai suatu sistem. Suatu sistem memiliki katrakteristik atau sifat tertentu,yaitu:

1. Komponen Sistem

   Suatu sistem yang terdiri atas bagian-bagian yang saling berkaitan dan bervariasi yang bersama-sama mencapai beberapa sasaran. Sebuah sistem bukanlah seperangkat unsur yang tersusun secara teratur, tetapi terdiri atas

unsur yang dapat dikenal dan saling melengkapi karena suatu maksud, tujuan dan sasaran. Batas Sistem

Batas Sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu ksatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

2. Lingkungan luar (*environment*)

   Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energy dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan menggangu kelangsungan hidup sistem.

3. Penghubung sistem (*interface*)

   Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran *(output*) untuk subsistem yang lainnya dengan melalui penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

4. Masukan Sistem (*input*)

   Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

5. Keluaran Sistem (*output*)

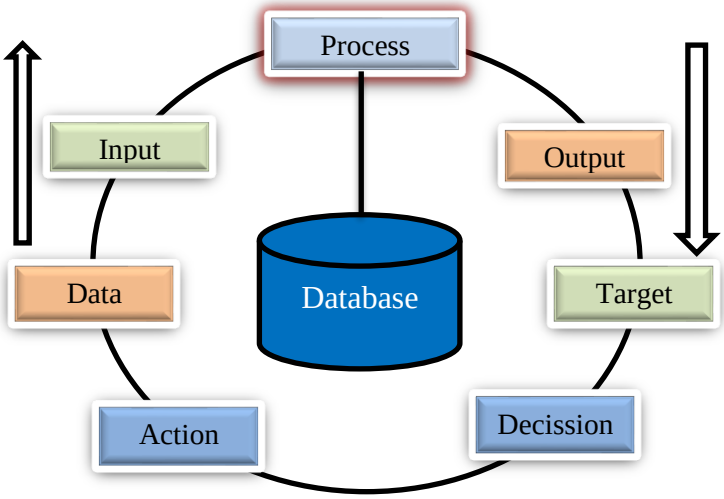
   Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keuaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem. Missal nya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

6. Sasaran Sistem (*goal*)

   Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak aka nada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.3 Informasi

Informasi (*information)* adalah data yang telah diolah menjadi suatu bentuk yang penting bagi sipenerima dan mempunyai nilai yang nyata atau dapat dirasakan mamfaatnya dalam keputusan-keputusan yang akan datang. *Output* informasi dari komputer digunakan oleh para Meneger, non Meneger, serta orang-orang dan organisasi-organisasi dalam lingkungan perusahaan. Yang lebih penting adalah mereka perlu mengerti informasi (*Information Literate*) seperti informasi apa yang mereka butuhkan, untuk apa informasi tersebut digunakan serta bagaimana mutu dan kualitas informasi yang dapat membantu mereka mengindentifikasi dan memecahkan masalah, untuk memutuskan tindakan yang akan diambil saat ini maupun yang akan dating. (Ali & Wangdra, 2010 : 10).

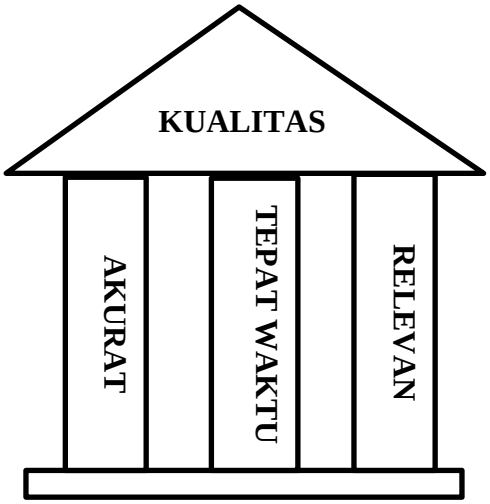


Gambar 2. 2 Siklus Informasi

Menurut (Winarno, 2006 : 6) informasi adalah data yang sudah diolah sehingga berguna untuk pembuatan keputusan. Data adalah representasi suatu objek. Misalnya seorang mahasiswa diwakili oleh nomor mahasiswa, maka nomor mahasiswa ini adalah data

Gambar 2. 3 Data diolah menjadi Informasi

Kualitas dari suatu sistem informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat waktunya (*timelines*) dan relevan (*relevance*). John Burch dan Gary Grudnitski menggambarkan kualitas dari informasi denganbentuk bangunan yang ditunjang oleh tiga buah pilar.



Gambar 2. 4 Pilar Kualitas Informasi

1. Akurat, berarti informasi harus bebas dari kesalahn-kesalahan dan tidak bisa menyesatkan. Akurat juga berarti informasi harus jeals mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusk informasi tersebut

2. Tepat waktu, berarti informasi yang dating pada penerima tidak boleh terlambat. Informasi yang suadah using tidak akan mempunyai nilai lai. Karena informasi merupakan landasan di dalam pengambilan keputusan.
3. Relevan, berarti informasi tersebut mempunyai mamfaat untuk pemakainya. Relevansi informasi untuk tiap-tipa orang satu denagan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila di tujukan kepada ahli teknik perusahaan.

2.1.4 Sistem Informasi

Sistem informasi (*information system*) merupakan suatu kumpulan dari komponen-komponen dalam suatu perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran sistem informasi. Dalam hal ini, TI merupakan salah satu komponen dalam perusahaan. Komponen-komponen yang lainnya adalah prosedur, struktur organisasi, sumberdaya manusia, produk, pelanggan, rekanan dan sebagainya.(Ali & Wangdra, 2010 : 13) .

Menurut (Tantra, 2012) Sistem Informasi adalah cara yang terorganisir untuk mengumpulkan, memasukkan, dan memproses data dan menyimpannya, mengelola, mengontrol dan melaporkannya sehingga dapat mendukung perusahaan atau organisasi untuk mencapai tujuan. Sistem informasi dapat bersifat formal dan informal, sistem informasi yang bersifat formal memang secara resmi memiliki tanggung jawab untuk menghasilkan informasi yang akurat, sedangkan sistem informasi informal adalah kebalikannya, berasal dari bagian-bagian

organisasi yang tidak secara resmi memberikan informasi, seperti misalnya bagian legal.

2.1.5 Komponen Sistem Informasi

(Muslihudin & Oktafianto, 2016: 14) John Burch dan Gary Grudnitski mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*) dan blok kendali (*controls block*).

1. Blok masukan (*input block)*

   Input mewakili data yang masuk ke daam sistem informasi. Input disini termasuk metode-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 yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (*output block*)

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

4. Blok teknologi (*technology block*)

   Teknologi merupakan “kotak alat” (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, minyimpan dan mengakses data, menghasilkan danmengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian utama, yaitu teknisi (*human atau brainware*), perangkat lunak (*sofware*) dan perangkat keras (*hardware*). Teknisi dapat berupa orang-orang yang mengetahui teknologi dan membuatnya dapat beropeasi.

5. Blok basis data (*database block*)

   Basis data (*database)* merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya,tesimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

6. Blok kendali (*controls block*)

   Banyak hal yang dapat merusk sistem informasi, seperti misalnya bencana alam, apai, temperature, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem

dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.1.6 *Internet*

(Hastanti, Eka, Indah, & Wardati, 2015) *Internet* adalah sebagai jaringan komputer yang sangat luas dan besar dan mendunia, menghubungkan pemakai komputer dari satu negara ke negara lain di seluruh dunia, dimana di dalamnya terdapat berbagai sumber informasi dan fasilitas–fasilitas layanan internet yaitu diantaranya:

1. *Browsing* atau *surfing* yaitu kegiatan *"berselancar*" di internet, kegiatan ini seperti layaknya berjalan-jalan di malsambil melihat-lihat ke toko-toko tanpa membeli apapun.
2. *Elektronik mail (E-mail)* fasilitas ini digunakan untuk berkirim surat dengan orang lain, tanpa mengenal batas, waktu, ruang bahkan birokrasi. *searching* yaitu kegiatan mencari data atau informasi tertentu di internet.
3. *Catting* fasilitas ini digunakan untuk berkomunikasi secara langsung dengan orang lain di Internet. Pada umumnya fasilitas ini sering digunakan untuk bercakap-cakap atau mengobrol di *internet World Wide Web* (WWW).
4. *Newsgroup* fasilitas ini digunakan untuk berkoferensi jarak jauh, sehingga anda dapat menyampaikan pendapat dan tanggapan dalam internet.
5. *Download* adalah proses mengambil file dari komputer lain melalui internet ke komputer kita.

6. *Upload* adalah proses meletakkan file dari komputer kita ke komputer lain melalui internet.
7. *Transfer protocol* (FTP) fasilitas ini digunakan untuk melakukan pengambilan arsip atau file secara elektronik atau transfer file dari satu komputer ke komputer lain di internet.
8. *Telnet* fasilitas ini digunakan untuk masuk ke sistem komputer tertentu dan bekerja pada sistem komputer lain.
9. *Ghoper* fasilitas ini digunakan untuk menempatkan informasi yang disimpan pada *internet server* dengan menggunakan hirarki.

Dari penjelasan diatas, *internet* merupakan kumpulan dari beberapa komputer yang terhubung dalam satu jaringan dan dapat diakses dari tempat yang sangat jauh.

2.1.7 *Website*

Website adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah website biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

Domain adalah nama unik yang dimiliki oleh sebuah institusi sehingga bisa diakses melalui intemet, misalnya lintau.com, yahoo.com, google.com, ephi.web.id, dan lain-lain. Istilah lain yang sering ditemui sehubungan dengan

website adalah *homepage*. *Homepage* adalah halaman awal sebuah domain (Yuhefizar, Mooduto, & Hidayat, 2009).

2.2 Tinjauan Teori khusus

Memuat dan menjelaskan teori khusus yang berhubungan dengan objek pembahasan sesuai dengan judul penelitian skripsi.

2.2.1 Rumah

Rumah adalah bangunan gedung yang berfungsi sebagai tempat tinggal yang layak huni, sarana pembinaan keluarga, cerminan harkat dan martabat penghuninya, serta aset bagi pemiliknya (UU No.1, 2011).

Rumah adalah bangunan untuk tempat tinggal, bangunan pada umumnya (seperti gedung); (https://kbbi.kemdikbud.go.id/, 2018)

Dalam arti umum, rumah adalah salah satu bangunan yang dijadikan tempat tinggal selama jangka waktu tertentu. Dalam arti khusus, rumah mengacu pada konsep-konsep sosial-kemasyarakatan yang terjalin di dalam bangunan tempat tinggal, seperti keluarga, hidup, makan, tidur, beraktivitas, rumah juga sebagai tempat untuk berumah tangga, tempat tinggal/alamat, lokasi tempat tinggal, bagian dari eksistensi individu/keluarga (terkait dengan status, tempat kedudukan, identitas), ruang untuk rekreasi, ruang yang digunakan untuk menjalin kehidupan keluarga, serta wadah sebagai batas privasi (Wikipedia, 2018).

2.2.2 Kontrakan (Sewa)

kontrak merupakan 1 perjanjian (secara tertulis) antara dua pihak dalam perdagangan, sewa-menyewa, dan sebagainya; 2 persetujuan yang bersanksi hukum antara dua pihak atau lebih untuk melakukan atau tidak melakukan kegiatan; kontrakan adalah yang dikontrak atau disewa (tentang rumah dan sebagainya); (https://kbbi.kemdikbud.go.id/, 2018).

Kontrak atau perjanjian adalah kesepakatan antara dua orang atau lebih mengenai hal tertentu yang disetujui oleh mereka. Penyewaan adalah sebuah persetujuan di mana sebuah pembayaran dilakukan atas penggunaan suatu barang atau properti secara sementara oleh orang lain. Barang yang dapat disewa bermacam-macam, tarif dan lama sewa juga bermacam-macam. Rumah umumnya disewa dalam satuan tahun, mobil dalam satuan hari, permainan komputer seperti *PlayStation* disewa dalam satuan jam (Wikipedia, 2018).

Pengertian sewa menurut kamus besar bahasa indonesia adalah pemakaian sesuatu dengan membayar uang sewa, uang yang dibayarkan karena memakai atau meminjamkan sesuatu, yang boleh pakai dengan membayar uang dengan uang. Pengertian penyewaan adalah suatu kesepakatan atau persetujuan dimana pihak yang satu menyanggupkan dirinya untuk menyerahkan suatu kebendaan kepada pihak lain, agar pihak ini dapat menikmatinya dalam jangka waktu tertentu, yang mana pihak yang belakang ini sanggup membayarnya (Prasetyo dan Sinta, 2014).

2.2.3 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak.

UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Rosa & Shalahuddin, 2013 : 133).

2.2.3.1 Sejarah *UML*

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Perkembangan aktif dari pemrograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian diikuti dengan perkembangan bahasa pemrograman berorientasi objek yang lainnya seperti *C objek, C++, Eiffel,* dan *CLOS.*

Sekitar lima tahun setelah Smalltalk berkembang, maka berkembang pula metode pengembangan berorientasi objek. Karena banyaknya metodologi-

metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep, seperti konsep *Object Modeling Technique (OMT)* dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities, Collaborators (CRC)* dari Rebecca Wirfs-Brock (1990), konsep pemikiran Ivar Jacobson, dan beberapa konsep lainnya dimana James R. Rumbaigh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama Rational Software Corporation menghasilkan bahasa yang disebut dengan *Unified Modeling Language (UML).*

Pada tahun 1996, *Object Management Group (OMG)* mengajukan proposal agar adanya standarisasi pemodelan berorientasi objek dan pada bulan September 1997 *UML* diakomodasi oleh *OMG* sehingga sampai saat ini *UML* telah memberikan kontribusinya yang cukup besar di dalam metodologi berorientasi objek dan hal-hal yang terkait di dalamnya.

Secara fisik, UML adalah sekumpulan spesifikasi yang dikeluarkan oleh OMG *(Object Management Group)*. UML terbaru adalah UML 2.3 yang terdiri dari 4 macam spesifikasi, yaitu *Diagram Interchange Specification*, UML *Infrastructure*, UML *Superstructure*, dan *Object Constraint Language* (OCL) (Rosa & Shalahuddin, 2013 : 136-140).

2.2.3.2 Diagram *UML*

UML adalah bahasa pemodelan yang standar untuk lingkungan berorientasi obyek, yang berisi notasi notasi grafis yang *relative* sudah

dibakukan (*open standard*). Sebagai sebuah sketsa, UML bisa berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek dari sistem. UML bisa juga berfungsi sebagai cetak biru karena sangat lengkap dan detil. Dengan cetak biru ini maka akan bisa diketahui informasi detil tentang coding program (*forward engineering*) atau bahkan membaca program dan menginterpretasikannya kembali ke dalam diagram *(reverse engineering*). Sebagai bahasa pemrograman, UML dapat menterjemahkan diagram yang ada di UML menjadi kode program yang siap untuk dijalankan (Lenti, 2014).

(Rosa & Shalahuddin, 2013 : 140), pada *UML* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. 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. *Structure diagram* terdiri dari *class diagram, object diagram, component diagram, composite structure diagram, package diagram dan deployment diagram.*
2. *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. Behavior diagram terdiri dari *Use case diagram, Aktivity diagram, State Machine System.*
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem. *Interaction diagram* terdiri dari *Sequence*

Diagram, Communication Diagram, Timing Diagram, Interaction Overview Diagram.

2.2.3.3 *Class* Diagram

Class Diagram adalah spesifikasi yang akan menghasilkan objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut atau properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode atau fungsi) (Isa & Hartawan, 2017).

(Rosa & Shalahuddin, 2013 : 141-147), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan *method* atau operasi. Berikut penjelasan atribut dan *method* :

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

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut:

1. Kelas main

   Kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas yang menangani tampilan sistem

   Kelas yang mendefinisikan dan mengatur tampilan ke pemakai.
3. Kelas yang diambil dari pendefinisian use case

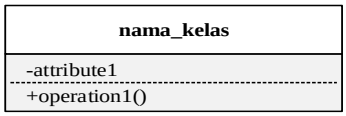
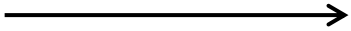
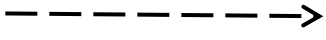
Kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case.

4. Kelas yang diambil dari pendefinisian data

   Kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

   Berikut adalah simbol-simbol yang ada pada diagram kelas :

Tabel 2. 1 Simbol- simbol yang ada pada diagram kelas

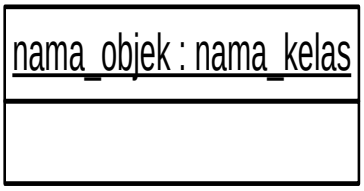
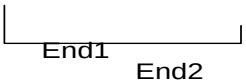
SIMBOL	DESKRIPSI
Kelas 	Kelas pada struktur sistem
Antarmuka/*interface* 	Sama dengan konsep *interface* dalam pemrograman berorientasi objek.
Asosiasi/*association* 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*.
Asosiasi berarah / *directed association* 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity*.
Ketergantungan/*dependency* 	Relasi antar kelas dengan makna kebergantungan antar kelas.
Agregasi / *aggregation* 	Relasi antar kelas dengan makna semua bagian (*whole-part*)

2.2.3.4 *Object* Diagram

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Diagram objek juga berfungsi untuk mendefinisikan contoh nilai atau isi dari atribut tiap kelas (Rosa & Shalahuddin, 2013 : 147).

Diagram objek menunjukkan sekumpulan objek dan keterhubungannya. Diagram ini menunjukkan potongan statik dari instan-instan yang ada di diagram kelas. Diagram ini untuk memperlihatkan satu prototipe atau kasus tertentu yang mungkin terjadi (Sopiah, 2012).

Tabel 2. 2 Simbol-simbol Yang ada pada diagram objek

SIMBOL	DESKRIPSI
Objek 	Objek dari kelas yang berjalan saat sistem dijalankan
Link 	Relasi antar objek

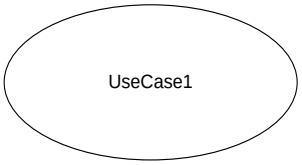
2.2.3.5 *Use Case* Diagram

Use Case diagram merupakan pemodelan untuk kelakuan *(behavior)* sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

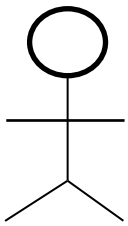
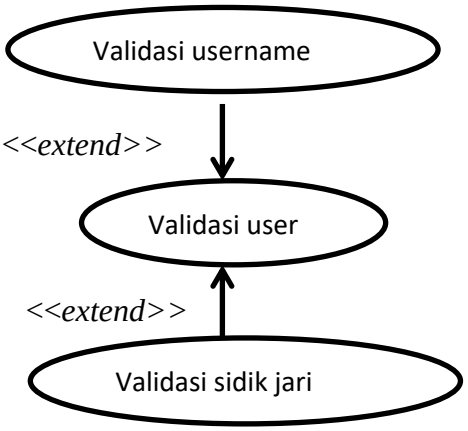
Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua Hal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *use case* (Rosa & Shalahuddin, 2013 : 155-158) :

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari actor adalah gambar orang, tapi actor belum tentu merupakan orang.
2. *Use Case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor.

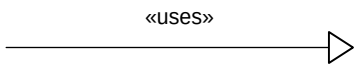
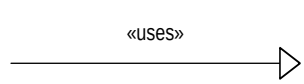
Tabel 2. 3 Simbol-simbol yang ada pada diagram *use case*

SIMBOL	DESKRIPSI
Use Case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau *actor*; biasanya dinyatakan dengan m enggunakan kata kerja di awal frase nama *use case*

Tabel 2. 3 Lanjutan simbol-simbol yang ada pada diagram *use case*

SIMBOL	DESKRIPSI
Aktor/*actor* 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi. Biasanya dinyatakan dengan menggunakan kata benda di awal frase nama *actor*
Asosiasi/*association*	Komunikasi antar *actor* dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan actor
Ekstensi/*extend*	Relasi *use case* tambahkan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan;  Arah panah mengarah pada *use case* yang ditambahkan.

Tabel 2. 3 Lanjutan simbol-simbol yang ada pada diagram *use case*

SIMBOL	DESKRIPSI
Generalisasi/*generalization* 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya. Arah panah mengarah pada *use case* yang menjadi generalisasi (umum).
Menggunakan/*include/user* **<<include>>**  <<uses>>	Relasi *use case* tambahkan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini. Ada dua sudut pandang yang cukup besar mengenai include di *use case* : • *Include* berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan • *Include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan.

2.2.3.6 *Activity* Diagram

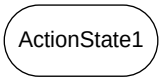
Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan actor, jadi aktivitas yang dapat dilakukan oleh sitsem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut (Rosa & Shalahuddin, 2013 : 161-162).

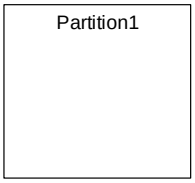
1. Rancangan prose bisnis di mana setiap urutan aktivitas yang digambarkanmerupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktiitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian di mana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Berikut adalah Simbol-simbol yang ada pada *activity* diagram:

Tabel 2. 4 Simbol-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.

Tabel 2. 4 Lanjutan Simbol-simbol *Activity* Diagram

SIMBOL	DESKRIPSI
Percabangan/*decision*	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan/join	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.

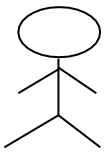
2.2.3.7 *Sequence* Diagram

(Rosa & Shalahuddin, 2013 : 165-167), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan *massage* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya diagram sekuen yang harus

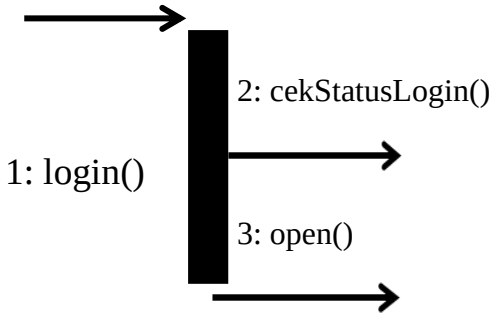
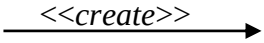
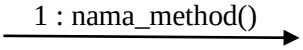
digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup dalam diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banayak.

Berikut adalah Simbol-simbol yang ada pada *sequence* diagram:

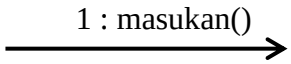
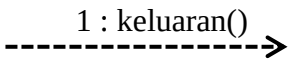
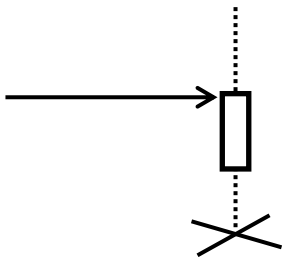
Tabel 2. 6 Simbol-simbol diagram *sequence*

SIMBOL	DESKRIPSI
Aktor  Atau Aktor	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, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dalam menggunakan kata benda diawal frase nama aktor.
Garis Hidup/ *lifetime*	Menyatakan kehidupan suatu objek
Objek Nama objek : nama kelas	Menyatakan objek yang berinteraksi pesan

Tabel 2. 6 Lanjutan Simbol-simbol diagram *sequence*

SIMBOL	DESKRIPSI
Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi, semuanya yang terhubungdengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya :  Maka cekStatusLogin() dan open() dilakukan didalam metode login(). Aktor tidak memiliki waktu aktif.
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 operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelasobjek yang berinteraksi.

Tabel 2. 6 Lanjutan Simbol-simbol diagram *sequence*

SIMBOL	DESKRIPSI
Pesan Tipe *send* 	Menyatakan bahwa suatu objek mengirimkan data/masukkan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirimi.
Pesan Tipe *Return* 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
Pesan Tipe *Destroy* 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada *create* maka ada *destroy*.

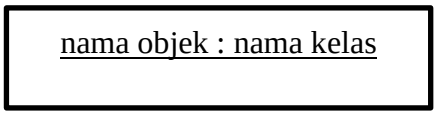
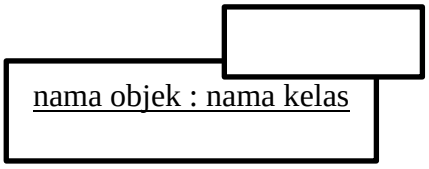
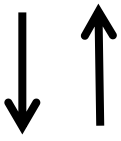
2.2.3.8 *Communication* Diagram

Diagram Komunikasi menggambarkan interaksi antarobjek/bagian dalam bentuk urutan pengiriman pesan. Diagram komunikasi mengelompokan *message* pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram komunikasi yang dituliskan adalah operasi/metode yang dijalankan antara objek yang satu dan objek lainnya secara keseluruhan, oleh karena itu dapat diambil dari jalannya interaksi pada semua diagram sekuen. Penomoran metode dapat

dilakukan berdasarkan urutan dijalankannya (Rosa & Shalahuddin, 2013 : 168-169).

Berikut simbol-simbol yang ada pada *communication* diagram:

Tabel 2. 7 Simbol-simbol pada *communication* diagram

SIMBOL	DESKRIPSI
Objek 	Objek yang melakukan interaksi pesan,
Link 	Relasi antar objek yang menghubungkan objek satu dengan yang lainnya atau dengan dirinya sendiri. 
Arah Pesan / *stimulus* 	Arah pesan yang terjadi, jika pada suatu *link* ada dua arah pesan yang berbeda maka arah juga digambarkan dua arah pada dua sisi *link* 

2.2.4 Flowchart

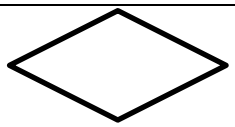
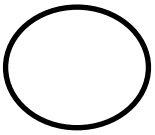
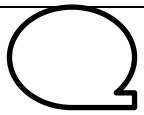
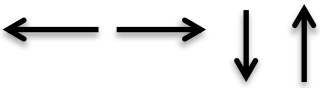
Bagian alir program (*Flowchart*) adalah bagian yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir".(Drs. Katen Lumbanbatu & Novriyeni, S.Kom., 2013).

Bagan alir program (Program *Flowchart*) merupakan bagan yang menjelaskan secara rinci, langkah-langkah dari proses program. Bagan alir program disebut dari derivikasi bagan alir sistem (Toibah Umi Kalsum1, 2012).

Flowchart atau bagan alir adalah suatu bagan yang berisi simbol-simbol grafis yang menunjukkan arah aliran kegiatan dan data-data yang dimiliki program sebagai suatu proses eksekusi. *Flowchart* pada umumnya menggambarkan suatu kegiatan dari interaksi yang ada dalam sistem, melalui proses (*input*) yaitu kegiatan atau masukan yang terjadi diawal sistem kemudian menggambarkan tahap proses yaitu kegiatan yang berlangsung setelah adanya proses (*input)* yang terjadi kepada sistem kemudian adanya pengolah yang digambarkan melalui bagan *flowchart* dengan simbol-simbol yang telah ada pada aturan *flowchart diagrams,* kemudian proses *output* yaitu keluaran yang terjadi setelah proses *Input* dan *process* yang secara keseluruhan diagram *flowchart* merepresentasiakan seluruh kegiatan atau aktivitas yang terjadi dalam sistem melalui bagan-bagan tertentu yang dapat dilihat secara visual. Simbol-simbol dalam *flowchart* itu sendiri mempunyai arti yang berbeda setiap simbol sehingga dalam pemakain atau penggunaan simbol *flowchart* harus menyesuaikan dengan nama kegiatan atau aktivitas yang akan dilakukan atau yang akan terjadi dalam sistem (Rasim, Setiawan, & Rahman, 2008).

Bentuk-bentuk yang digunakan dalam pembuatan flowchart mempunyai arti-arti khusus, yaitu sebagai berikut:

Tabel 2. 8 Simbol Flowchart

Simbol Flochart	**Keterangan**
	Memulai atau Mengakhiri Program
	Proses (Menulis atau Menjalankan) Program
	Masukan atau Keluaran Pengambilan
	Pengambilan Keputusan atau Pengujian Program
	Penghubung Program Magnetic
	Magnetic Tape Magnetic
	Magnetic Disk Arah
	Arah aliran
	Tampilan/Penyajian Hasil Pemprosesan Data

2.2.5 DFD (*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah diagram yang menggunakan notasi-notasi untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan.(Giyanmandiri, 2016)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data, dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data yang tersimpan, dan proses yang dikenakan pada data tersebut.(Afyenni, 2014)

Ada empat buah simbol pada DFD, yang masing-masingnya digunakan untuk mewakili (Afyenni, 2014) :

a. External entity (kesatuan luar) atau *boundary* (batas sistem), digunakan untuk menyatakan: suatu kantor, departemen atau divisi dalam perusahaan tetapi di luar sistem yang dikembangkan; orang atau sekelompok orang di organisasi tetapi di luar sistem yang sedang dikembangkan; suatu organisasi atau orang yang berada di luar organisasi misal: langganan, pemasok; sistem informasi yang lain di luar sistem yang sedang dikembangkan; sumber asli suatu transaksi; penerima akhir dari suatu laporan yang dihasilkan oleh sistem.
b. Data *flow* (arus data), digunakan untuk menunjukkan arus dari data yang dapat berupa: masukan untuk sistem ataupun hasil dari proses sistem. Arus data

sebaiknya diberi nama yang jelas dan mempunyai arti. Didalam menggambarkan arus data di DFD perlu diperhatikan beberapa konsep berikut:

1. Konsep paket dari data (*packet of data*). Bila dua atau lebih data mengalir dari suatu sumber yang sama ketujuan yang sama, maka dianggap sebagai suatu arus data tunggal.
2. Konsep arus data menyebar. Menunjukkan sejumlah tembusan dari arus data yang sama dari sumber yang sama ketujuan berbeda.
4. Konsep arus data mengumpul *(converging data flow*) Menunjukkan beberapa arus data yang berbeda bergabung bersama- sama menuju ke tujuan yang sama.
5. Konsep sumber dan tujuan arus data Semua arus data harus dihasilkan dari suatu proses atau menuju ke suatu proses.

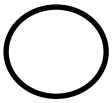
c. Proses (*process*), digunakan untuk menunjukkan kegiatan atau kerja yag dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang keluar dari proses. Suatu proses harus menerima arus data dan menghasilkan arus data.

d. Data *store* (simpanan data), digunakan untuk menunjukkan simpanan dari data yang dapat berupa: suatu file atau database di sistem komputer; suatu arsip atau catatan manual; suatu kotak tempat data di meja seseorang; suatu tabel acuan manual; suatu agenda atau buku.

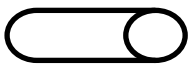
Diagram alir data merupakan model dari sistem untuk menggambarkan sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram alir data adalah memudahkan pemakai atau *user* yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan.

Berikut ini merupakan gambar *data flow diagram* yang umum digunakan dalam perancangan sistem:

Tabel 2. 9 Simbol DFD 1

Simbol	Nama	Keterangan
	Terminal	Awal atau akhir
	Input/Output	Mempresentasikan input data atau output data yang diproses atau informasi
	Penghubung	Keluar atau masuk ke bagian yang lain dalam halaman yang sama
	Anak panah	Mempresentasikan alur kerja
	Penjelasan	Dugunakan untuk komentar tambahan
	Proses	Mempresentasikan operasi
	Keputusan	Keputusan dalam program
	Predefined proses	Rincihan operasi yang berada di tempat lain
	Preparation	Pemberian harga awal

. **Tabel 2. 9** Symbol DFD 2 Lanjutan

Simbol	Nama	Keterangan
	Dokumen	I/O dalam format yang dicetak
	Manual Operaion	Operasi manual
	Online Storage	I/O yang mggunakan penyimpan akses lngsung
	Communication link	Transmisi data melalui chanel koneksi
	Punched Tape	I/O yang menggunakan pita kertas penghubung
	Punched Card	I/O yang menggunakan pita kartu penghubung
	Magnetic Tape	I/O yang menggunakan pita magnetic tape
	Magnetic Disk	I/O yang menggunakan magnetic disk
	Magnetic Drum	I/O yang menggunakan Magnetic Drum

2.2.6 HTML (*Hyper Text Markup Language*)

HTML (*Hyper Text Markup Language*) bisa disebut bahasa paling dasar dan penting yang digunakan untuk menampilkan dan mengelola tampilan pada halaman website. HTML digunakan untuk manampilkan berbagai informasi didalam sebuah penjelajah *web* internet dan *formatting hypertext* sederhana yang ditulis kedalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi (Saputra, 2012).

Sejarah html berawal pada tahun 1980 ketika IBM berniat untuk membuatkan suatu bahasa kode untuk menggabungkan teks dengan pemformatan agar mengenali elemen dokumen. Bahasa yang menggunakan tanda-tanda ini dinamakan *Markup Language*. Namin pihak IBM memberi nama *Generalized Markup Language* (GML).

Dokumen html memiliki struktur yang harus kita ikuti aturan pembuatannya. Beberapa elemen-elemen htm yang wajib ada pada html adalah berikut:

1. Elemen HTML

   Elemen HTML merupakan tag dasar apabila kita ingin memulai suatu dokumen html. Secara logika, jika kita menemukan tag ini, berarti secara jelas dapat didefenisikan bahwa dokumen ini merupakan perintah suatu dokumen html. Tag ini merupakan perintah wajib bagi pemrprogram web untuk menuliskan tag pertama dalam dokumen html

   <html> …..<html>

2. Elemen *Head*

Head merupakan tag berikutnya setelah elemen html, yang berfungsi untuk menuliskan keterangan tentang dokumen web yang akan ditampilkan. Elemen ini nantinya akan diakhiri dengan tanda penutup <head>. Contoh:

<html> </head>
<head> </html>

3. Elemen *Title*

Elemen *title* merupakan suatu elemen yang harus dituliskan didalam elemen *head* yang digunakan untuk memberikan judul/ informasi pada *caption browser* web tentang topik/ tema atau judul dari suatu dokumen web yang ditampilkan pada browser. Berikut struktur penggunaannya:

```
<html>
    <head>
        <title>judul</title>
    </head>
</html>
```

4. Elemen *Body*

Merupakan bagian utama dalam dokumen web. Jika kita ingin menampilkan suatu teks atau informasi atau yang dikenal dengan sebutan konten, maka kita harus meletakkan teks tersebut pada elemen *body*. Struktur elemennya sebagai berikut:

```
<html>
<head>
        <title>judul</title>
</head>
        <body>isi konten</body>
</html>
```

2.2.7 CSS *(Cascading Style Sheet)*

CSS atau yang memiliki kepanjangan *Cascading Style Sheet*, merupakan suatu bahasa pemrograman *web* yang digunakan untuk memperindah tampilan halaman *website* (situs), mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* akan lebih rapi, terstruktur, dan seragam". (Sagita & Sugiarto, 2016).

CSS merupakan singkatan dari *Cascading Style Sheet*. *CSS* biasa digunakan dalam dokumen *HTML* untuk menciptakan suatu *style* yang dipakai untuk mengatur penampilan elemen *HTML*. Dengan menggunakan *style*, suatu elemen dapat diformat dengan fitur yang jauh lebih kaya daripada yang disediakan oleh elemen *HTML* itu sendiri. Sebagai contoh, pengaturan seperti warna tulisan bisa ditangani melalui *style* tanpa melibatkan tag *HTML* yang berfungsi untuk mengatur warna (Maudi, Nugraha, & Sasmito, 2014).

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh teks, warna tabel, ukuran border, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, margin kiri, kanan, atas, bawah, dan parameter lainnya (Muarie, 2015).

2.2.8 PHP *(Personal Home Page)*

PHP adalah pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. PHP merupakan singkatan dari *Hypertext Preprocessor*, dan merupakan bahasa yang disertakan dalam dokumen HTML

sekaligus bekerja di sisi *server* (*server-side HTML embedded scripting*) (Sagita & Sugiarto, 2016).

Beberapa kelebihan dari PHP itu sendiri antara lain:

a. Mudah dipelajari.
b. Mampu lintas *platform*.
c. *Free* (gratis) bersifat *open source*
d. PHP memiliki tingkat akses yang cepat
e. Didukung oleh beberapa macam *web* server.
f. Mendukung database.

PHP adalah bahasa pemrograman skrip sederhana yang digunakan untuk pemrosesan HTML *Form* di dalam halaman *web*. Strukturnya sangat sederhana sehingga PHP dapat dengan mudah dipelajari programmer pemula bahkan orang tanpa latar belakang Teknologi Informasi. Hal inilah yang menyebabkan PHP sangat cepat populer di kalangan pengembang aplikasi *web*. Membuat program menggunakan PHP itu mudah, cukup sediakan saja sebuah program editor teks sederhana untuk menuliskan programnya, seperti *Notepad* (*Windows*) dan *vi editor* (*Linux*), atau program editor yang lebih *advance*, seperti *EditPlus*, *Notepad++*, atau *Dreamweaver* (Hastanti et al., 2015).

2.2.9 XAMPP

XAMPP merupakan paket PHP berbasis *open source*. Informasinya dapat diperoleh di website resminya: http://www.apachefriends.com. XAMPP membantu memudahkan dalam mengembangkan aplikasi berbasis PHP. XAMPP

mengkombinasikan beberapa paket *software* berbeda kedalam satu paket. Adapun lisensi masing-masing paket *software* tersebut dapat ditemukan didirektori \xampp\licence.

XAMPP menyediakan antar muka *control panel* tersendiri yang dapat digunakan untuk menjalankan semua *service* (paket *software* pendukung) yang telah terinstal. Pada sistem operasi windows, *control panel* dapat diakses melalui menu [Start]→ [Program] →[Apachefriends] → [xampp] →[control xampp server panel].

Pada web server (lokal komputer, tidak di server internet sesungguhnya) pada XAMPP, akan menyediakan satu *folder* kerja yang bernama htdocs. Pada paket ini, *folder* kerja tersebut dapat ditemukan pada *subfolder* C:\..\XAMPP (sesuai lokasi dimana menyimpan hasil instalasinya) (Maudi et al., 2014).

2.2.10 MySQL (*My Structure Query Language*)

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL yang *multithread*, multi-*user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL adalah implementasi dari manajemen basis data relasional (RDBMS).

Pada saat ini MySQL merupakan basis data server yang sangat terkenal di dunia, semua itu karena bahasa dasar yang digunakan untuk mengakses basis data yaitu SQL (*Structure Query Language*). Dengan menggunakan SQL, proses pengaksesan basis data lebih *user-friendly* dibandingkan dengan yang lain, misalnya *dBAse* atau *clipper* karena mereka masih menggunakan perintah-perintah pemrograman murni (Maudi et al., 2014).

2.2.11 *JavaScript*

JavaScript adalah bahasa skrip yang ditempatkan pad kode HTML dan diproses pada sisi klien. Dengan adanya bahasa ini maka kemampuan dokumen HTML menjadi lebih luas. Sebagai contoh, digunakan untuk validasi masukan pada formulir sebelum diproses ke tahap selanjutnya. Bisa untuk membuat permainan interaktif dan juga bisa untuk menambah desain *web* (Muarie, 2015).

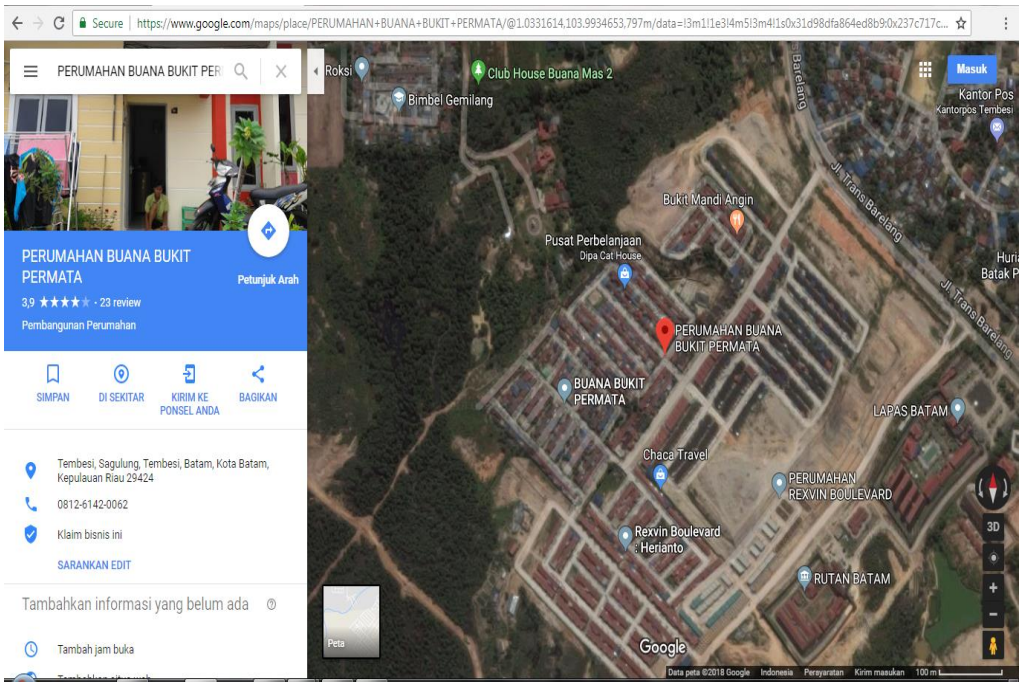
JavaScript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, bahasa ini adalah bahasa skrip pertama untuk web. Bahasa pemrograman ini memberikan kemampuan tambahan terhadap bahasa HTML dengan mengizinkan pengeksekusian perintah-perintah di sisi client, yang artinya di sisi browser bukan di sisi server web. JavaScript mampu mengakses Document Object Model serta melakukan manipulasi terhadapnya (Yenni & Shamir, 2012).

2.2.12 *Google Maps*

Google Map Service adalah sebuah jasa peta global *virtual* gratis dan *online* yang disediakan oleh perusahaan *Google. Google Maps yang* dapat ditemukan di alamat *http://maps.google.com. Google Maps* menawarkan peta yang dapat diseret dan gambar satelit untuk seluruh dunia.*Google Maps* juga menawarkan pencarian suatu tempat dan rute perjalanan.

Google Maps API adalah sebuah layanan (*service*) yang diberikan oleh *Google* kepada para pengguna untuk memanfaatkan *Google Map* dalam mengembangkan aplikasi. *Google Maps API* menyediakan beberapa fitur untuk

memanipulasi peta, dan menambah konten melalui berbagai jenis *services* yang dimiliki, serta mengijinkan kepada pengguna untuk membangun aplikasi *enterprise* di dalam websitenya (Soepomo, 2013).



Gambar 2. 5 Tampilan *Google Maps*

Pengguna dapat memanfaatkan layanan-layanan yang ditawarkan oleh *Google Maps* setelah melakukan registrasi dan mendapatkan *Google Maps API Key*.*Google* menyediakan layanan ini secara gratis kepada pengguna di seluruh dunia. *GPS* atau *Global Positioning System*, merupakan sebuah alat atau sistem yang dapat digunakan untuk menginformasikan penggunanya dimana lokasinya berada (secara global) di permukaan bumi yang berbasiskan satelit. Data dikirim dari satelit berupa sinyal radio dengan data digital. Dimanapun pengguna tersebut berada, maka *GPS* bisa membantu menunjukan arah. Layanan *GPS* ini tersedia gratis (Soepomo, 2013).

Awalnya *GPS* hanya digunakan hanya untuk kepentingan militer, tapi pada tahun 1980-an dapat digunakan untuk kepentingan sipil. *GPS* dapat digunakan dimanapun juga dalam 24 jam. Posisi unit *GPS* akan ditentukan berdasarkan titik-titik koordinat *latitude* dan *longitude*, untuk bisa mendapatkan posisi *latitude* dan *longitude* dapat dikunjungi di *http://www.mapcoordinates.net:*

a. Latitude

Latitude atau garis lintang adalah adalah garis maya yang melingkari bumi ditarik dari arah barat hingga ke timur atau sebaliknya , sejajar dengan equator (garis khatulistiwa). Garis lintang terus melingkari bumi, dari equator hingga ke bagian kutub utara dan kutub selatan bumi.

b. Longitude

Longitude atau garis bujur adalah garis maya yang ditarik dari kutub utara hingga ke kutub selatan atau sebaliknya. Dengan pengetahuan seperti itu berarti derajat antar garis bujur semakin melebar di daerah khatulistiwa dan makin menyempit di daerah kutub. Jika pada Garis Lintang, daerah yang dilalui garis khatulistiwa (*equator*) dianggap sebagai nol derajat, untuk Garis Bujur, tempat yang dianggap sebagai nol derajat adalah garis dari kutub utara ke kutub selatan yang tepat melintasi kota Greenwich di Inggris.

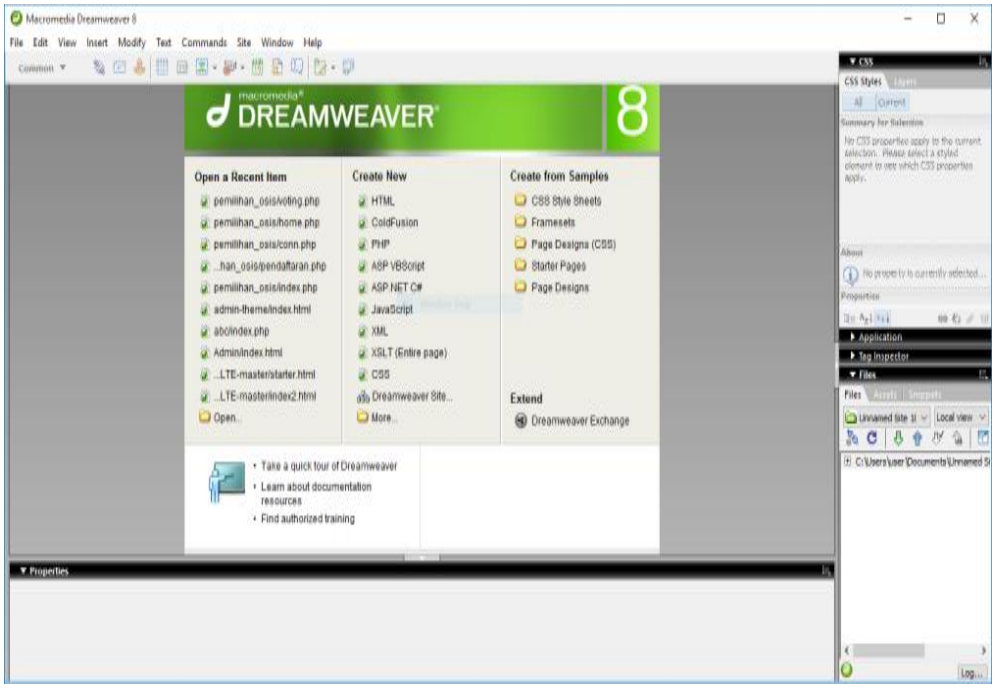
2.2.13 Dreamweaver

Adobe Dreamweaver merupakan program penyunting halaman web keluaran Adobe Systems yang dulu dikenal sebagai Macromedia Dreamweaver

keluaran Macromedia. Program ini banyak digunakan oleh pengembang *web* karena fitur-fiturnya yang menarik dan kemudahan penggunaannya.

Versi terakhir Macromedia Dreamweaver sebelum Macromedia dibeli oleh *Adobe Systems* yaitu versi 8. Versi terakhir Dreamweaver keluaran *Adobe Systems* adalah versi 10 yang ada dalam Adobe Creative Suite 4 (sering disingkat Adobe CS4) (Maudi et al., 2014).

Menurut jurnal (Destiningrum & Adrian, 2017), Adobe Dreamweaver adalah : "aplikasi desain dan pengembangan web yang menyediakan editor WYSIWYG visual (bahasa sehari-hari yang disebut sebagai *Design view*) dan kode editor dengan fitur standar seperti *syntax highlighting, code completio*n, dan *code collapsing* serta fitur lebih canggih seperti *real-time syntax checking* dan *code introspection* untuk menghasilkan petunjuk kode untuk membantu pengguna dalam menulis kode".



Gambar 2. 6 Tampilan Macromedia Dreamweaver

2.3 Penelitian terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian penulis. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Untuk memudahkan pemahaman terhadap bagian ini, dapat dilihat pada tabel berikut ini:

Tabel 2. 11 Tabel Penelitian Terdahulu

No	Tahun/ ISSN	Peneliti	Judul Penelitian	Metode	Kesimpulan	Penerbit
1	2015/ 1693-590x	Didit Gunawan, Ernes Cahyo Nugroho	Sistem Informasi Sewa Rumah Kost Dan Rumah Kontrakan Berbasis Web Di Surakarta	Metode *Waterfall*	Membantu dalam memberikan informasi kepada konsumen tentang kamar yang sudah terisi dan memungkinkan pelanggan mendapatkan detail informasi tentang kos yang di cari.	STMIK AUB Surakarta

Tabel 2. 11 Lanjutan Tabel Penelitian Terdahulu

No	Tahun / ISSN	Peneliti	Judul Penelitian	Metode	Kesimpulan	Penerbit
2	2013/ 2338-6304	Guridno Adi Saputro, Amir Hamzah, Joko Triyono	Sistem Informasi Pengolahan Data Rumah Kos Mengguna kan *Framework Codeignite r* Di Kelurahan Demangan Kecamatan Gondokusu man Yogyakarta	*Model View Controller*	Sistem ini memilik fitur pencarian yang mudah dan proses pemesanan Kos yang cepat.	Teknik Informatika Institut Sains & Teknologi AKPRIND Yogyakarta
3	2016/ 2442-4943	Dadi Rosadi, Feby Oktarista Andriawan	Aplikasi Sistem Informasi Pencarian Tempat Kos Di Kota Bandung Berbasis *Android*	Metode *Waterfall*	Pencarian informasi kost menjadi lebih mudah dan efisian serta mendetail tanpa harus datang langsung kelokasi. Aplikasi mudah digunakan bagi para pemilik kos yang ingin mempromosikan kosannya.	STMIK Mardira Indonesia, Bandung

Tabel 2. 11 Lanjutan Tabel Penelitian Terdahulu

No	Tahun / ISSN	Peneliti	Judul Penelitian	Metode	Kesimpulan	Penerbit
4	(2250-3153, 2014)	Henry Peter Gommans, George Mwenda Njiru, Arphaxad Nguka Owange	*Rental Housing Management System*	*waterfall model*	*the software can be used as an inventory system to provide a frame work that enables the mangers to make reasonable transactions made within a limited time frame.*	*Supply Chain Professional/Lecturer), Tradewinds Kenya Limited Scholar*
5	(2220-8488, 2015)	Sani, Kuzayet Shaibu, Gbadegesin, Job Taiwo	*A Study of Private Rental Housing Market in Kaduna Metropolis, Nigeria*	*Analyzed using methods descriptive statistics*	*Government should formulate favorable housing policy that will guarantee easy access to urban rental housing either through subsidized rent or rent control.*	*Department of Estate Management Obafemi Awolowo University Ile-Ife, Nigeria*