



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Menurut Sujatmiko (2012:156) mengenai pengertian komputer, “Komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi komunikasi”.

Menurut Blissmer dalam Jogiyanto (2005:1) mengenai pengertian komputer, “Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas berikut: menerima *input*, memproses *input* tadi sesuai dengan programnya, menyimpan perintah-perintah dan hasil dari pengolahan, menyediakan *output* dalam bentuk informasi”.

Dari beberapa definisi komputer diatas penulis menyimpulkan bahwa komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas berikut: menerima *input*, memproses *input* tadi sesuai dengan programnya, menyimpan perintah-perintah dan hasil dari pengolahan, menyediakan *output* dalam bentuk informasi yang bermanfaat dan dapat bekerja secara otomatis.

2.1.2. Pengertian Perangkat Lunak

Menurut Sukanto et al (2014:2) mengenai pengertian perangkat lunak, “Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”.

Menurut Sujatmiko (2013:201) mengenai pengertian perangkat lunak, “Perangkat lunak adalah istilah umum untuk data yang diinformasikan dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer”.



2.1.3. Pengertian Data

Menurut Kristanto (2011:7) mengenai pengertian data, “Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata.”

Menurut Sutabri (2012:1) mengenai pengertian data “Data merupakan bentuk jamak dari bentuk tunggal datum. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata”.

Menurut Sujatmiko (2012:76) mengenai pengertian data “Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi”.

Dari beberapa definisi diatas dapat disimpulkan bahwa data adalah sekumpulan fakta yang harus diolah lebih lanjut untuk menghasilkan suatu informasi.

2.1.4. Pengertian Sistem

Menurut Sutabri (2012:3) mengenai pengertian sistem, Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung, satu sama lain

Kristanto (2011:2), “Sistem merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (input) yang ditujukan kepada sistem tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (ouput) yang diinginkan.”

Jadi, sistem merupakan sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

2.2. Teori Judul

2.2.1. Pengertian Aplikasi

Asropudin (2013:7), “Aplikasi dibuat untuk mengerjakan atau menyelesaikan masalah-masalah khusus.”



Sujatmiko (2013:23), *Application* adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms. Word, Ms.Excel dan lain-lain.

Dari beberapa definisi diatas dapat disimpulkan bahwa aplikasi adalah perangkat lunak yang digunakan untuk mengerjakan masalah-masalah khusus.

2.2.2. Pengertian Pengolahan Data

Menurut Kristanto (2008:8) mengenai pengertian pengolahan data, “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Menurut Rasyad (2003:14), “Pengolahan data adalah proses untuk memperoleh data/angka ringkasan (summary figure) yang berasal dari sekelompok data mentah dengan menggunakan rumus tertentu”

2.2.3. Pengertian Rekam Medis

Menurut Jusuf mengenai pengertian rekam medis, “Rekam Medis adalah berkas yang berisikan catatan, dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan, dan pelayanan lain pada pasien pada sarana pelayanan kesehatan”.

Direktorat Jenderal Bina Pelayanan Medik (2006:11), Rekam medis diartikan sebagai keterangan baik yang tertulis maupun yang terekam tentang identitas, anamnesa, pemeriksaan fisik, laboratorium, diagnosa serta segala pelayanan dan tindakan medis yang diberikan kepada pasien, dan pengobatan baik yang dirawat inap, rawat jalan maupun yang mendapatkan pelayanan gawat darurat.

Direktorat Jenderal Bina Pelayanan Medik (2006:11), Sesuai dengan penjelasan Pasal 46 ayat (1) UU No.29 Tahun 2004 tentang praktik kedokteran disebutkan bahwa, yang dimaksud Rekam Medis adalah berkas yang berisikan catatan dan dokumen tentang identitas pasien, pemeriksaan, pengobatan, tindakan dan pelayanan lain yang telah diberikan kepada pasien. Dan yang dimaksud



dengan petugas adalah dokter atau dokter gigi atau tenaga kesehatan lain yang memberikan pelayanan kesehatan secara langsung kepada pasien.

Dari beberapa definisi diatas dapat disimpulkan bahwa rekam medis adalah berkas yang berisikan catatan dan dokumen identitas, anamnesa, pemeriksaan fisik, laboratorium, diagnosa serta segala pelayanan dan tindakan medis yang diberikan kepada pasien, dan pengobatan baik yang dirawat inap, rawat jalan maupun yang mendapatkan pelayanan gawat darurat.

2.2.4. Pengertian Pasien Rawat Jalan

Menurut Rahayu (2014:403), “Pasien rawat jalan adalah pasien yang hanya memperoleh pelayanan kesehatan tertentu dan tidak melakukan rawat inap atau opname”.

2.2.5. Pengertian Pasien Rawat Inap

Menurut Rahayu (2014:403), “Pasien rawat inap adalah pasien yang memperoleh pelayanan tinggal atau dirawat inap pada suatu unit kesehatan tertentu; pasien yang dirawat inap dirumah sakit”.

2.2.6. Pengertian Rumah Sakit Ibu dan Anak

Menurut Rahayu (2014:475), “Rumah Sakit Ibu dan Anak adalah rumah sakit yang khusus melayani pemeriksaan ibu-ibu hamil, ibu-ibu yang akan melahirkan, dan menangani kesehatan balita”.

2.2.7. Pengertian Judul Secara Keseluruhan

Aplikasi Pengolahan Data Rekam Medis Rawat Jalan dan Rawat Inap pada Rumah Sakit Ibu dan Anak Siti Mirza Palembang merupakan suatu perangkat lunak yang dikembangkan untuk difungsikan secara khusus dalam mengelola data pasien baik itu rawat inap maupun rawat jalan serta mampu membuat laporan bulanan jumlah pasien yang melakukan rawat inap dan rawat jalan di Rumah Sakit Ibu dan Anak Siti Mirza Palembang, sehingga pengolahan data pasien dapat



dilakukan secara terkomputerisasi dan terhubung komputer satu dengan komputer lain.

2.2.8. Teknik Pengumpulan Data

Menurut Ruslan (2010:22-30), pengumpulan data (*input*) merupakan suatu langkah dalam metode ilmiah melalui prosedur sistematis, logis, dan proses pencarian data yang valid, baik diperoleh secara langsung (*primer*) atau tidak langsung (*seconder*) untuk keperluan analisis dan pelaksanaan pembahasan (*proses*) suatu riset secara benar untuk menemukan kesimpulan, memperoleh jawaban (*output*) dan sebagai upaya untuk memecahkan suatu persoalan yang dihadapi oleh peneliti. Berdasarkan cara perolehannya data terbagi menjadi 2 (dua), yaitu:

a. Data Primer (*Primary Data*)

Data primer merupakan data yang diperoleh secara langsung dari objek penelitian perorang, kelompok, dan organisasi. Metode pengumpulan data primer dilakukan dengan cara sebagai berikut:

1. Wawancara (*Interview*)

Wawancara merupakan salah satu teknik pengumpulan data secara primer dalam metode survei melalui daftar pertanyaan yang diajukan secara lisan terhadap reponden (subjek). Teknik wawancara dapat dilakukan dengan tatap muka (*face to face interviews*) dan melalui saluran telepon (*telephon interviews*).

Penulis melakukan wawancara dengan pegawai-pegawai di RSIA Siti Mirza Palembang terhadap apa yang dilakukan dan dihasilkan, maupun hal-hal yang ingin penulis ketahui dari sistem kerja disana, seperti menanyakan profil dan gambaran umum perusahaan, serta sistem apa yang dibutuhkan perusahaan saat ini.

2. Observasi (*Observation*)

Observasi/pengamatan merupakan metode pengumpulan data primer dengan memperolehnya secara langsung dari sumber penelitian.



Penulis mengamati atas apa yang dikerjakan oleh setiap satuan kerja pada RSIA Siti Mirza Palembang secara langsung.

b. Data Sekunder (*Secondary Data*)

Data sekunder merupakan data penelitian yang diperoleh secara tidak langsung melalui media perantara (dihasilkan pihak lain) atau digunakan oleh lembaga lainnya yang bukan merupakan pengolahnya, tetapi dapat dimanfaatkan dalam suatu penelitian tertentu. Data sekunder pada umumnya berbentuk catatan atau laporan data dokumentasi oleh lembaga tertentu yang dipublikasikan.

Penulis mengambil data yang dibutuhkan melalui buku pedoman perusahaan, seperti sejarah perusahaan, visi dan misi, nilai-nilai utama, data peserta kerja praktek, dan data lainnya yang dibutuhkan penulis dalam penyusunan laporan.

2.3. Teori Khusus

2.3.1. Pengertian Pemrograman Berorientasi Objek

Menurut Sukamto et al (2014:100), “Berorientasi Objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya”.

2.3.2. Konsep Dasar Berorientasi Objek

Beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek adalah sebagai berikut:

1. Kelas (*Class*)

Kelas adalah kumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi statik dan himpunan objek yang sama yang mungkin lahir atau diciptakan dan kelas tersebut. Sebuah kelas akan mempunyai sifat (atribut), kelakuan (metode), hubungan (*relationship*).



2. Objek (*Object*)

Objek adalah abstraksi dan sesuatu yang mewakili dunia nyata seperti benda, manusia, satuan organisasi, tempat, kejadian, struktur, status atau hal-hal lain yang bersifat abstrak. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan.

3. Metode (*Method*)

Operasi atau metode atau *method* pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuah kelas boleh memiliki lebih dari satu metode atau operasi. Metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri. Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek.

4. Atribut (*Attribute*)

Atribut dari sebuah kelas adalah variable global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis dan sebagainya.

5. Abstraksi (*Abstraction*)

Prinsip untuk mempresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

6. Perwarisan (*Inheritance*)

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dari dirinya.

7. Antarmuka (*Interface*)

Antarmuka atau *interface* sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa ini. Deklarasi metode pada sebuah *interface* dapat diimplementasikan oleh kelas lain,.



8. Reusability

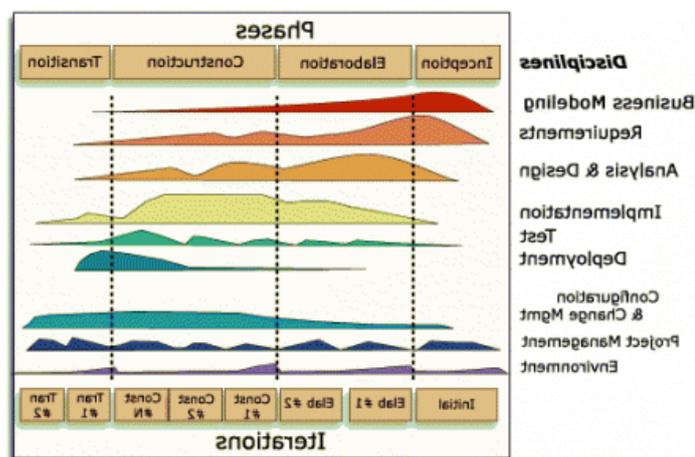
Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainnya yang melibatkan objek tersebut. Misalkan dalam sebuah aplikasi peminjaman buku diperlukan kelas anggota, maka ketika membuat aplikasi penyewaan VCD, kelas anggota ini bisa digunakan kembali dengan sedikit perubahan untuk aplikasi penyewaan VCD tanpa harus membuat dari awal kembali.

9. Generalisasi dan Spesialisasi

Menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus. Misalnya kelas yang lebih umum (generalisasi) adalah kendaraan darat dan kelas khususnya (spesialisasi) adalah mobil, motor, dan kereta.

2.3.3. Metodologi RUP (*Rational Unified Process*)

Menurut Sukamto et al (2014:125), “RUP (*Rational Unified Process*) adalah pengembangan perangkat lunak yang dilakukan berulang-ulang, fokus pada arsitektur (*architecture-centric*), lebih diarahkan berdasarkan penggunaan kasus (*use case driven*). RUP merupakan proses rekayasa perangkat lunak dengan pendefinisian yang baik (*well defined*) dan penstrukturan yang baik (*well structured*). RUP menyediakan pendefinisian struktur yang baik untuk alur hidup proyek perangkat lunak”.



Gambar 2.1 Arsitektur *Rational Unified Process*

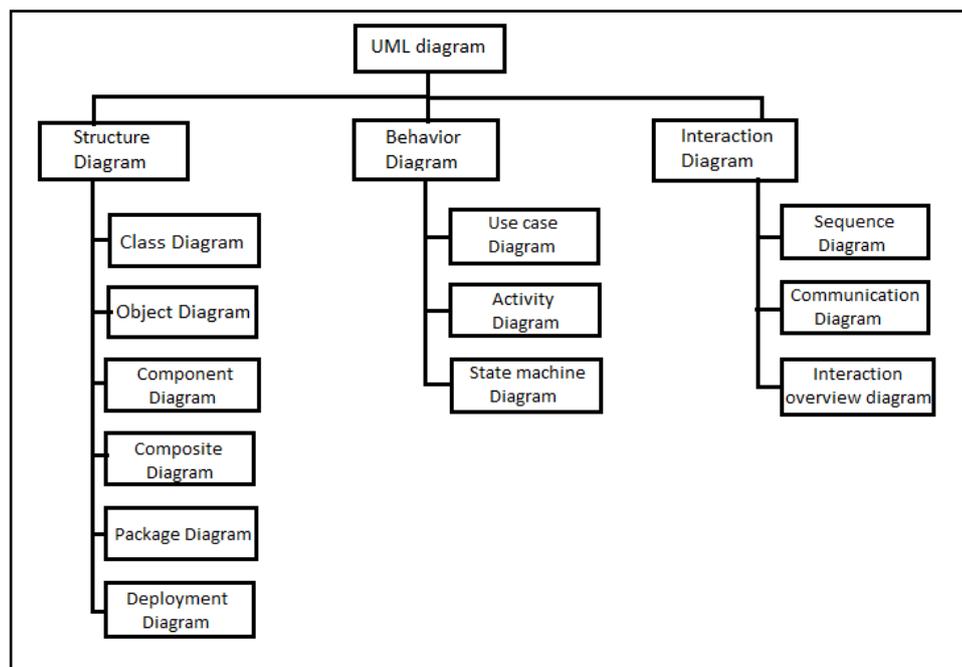


2.3.4. *Unified Modelling Language (UML)*

Menurut Sukamto et al (2014:133), “*UML (Undefined Modelling Language)* 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.”

2.3.5. *Macam-macam Diagram Unified Modeling Language*

Menurut Sukamto et al (2014:140), Pada *UML* terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar di bawah ini:



Gambar 2.2 Diagram UML

(*Sumber: Sukamto et al, 2014:140*)

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.



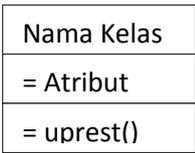
2. *Behavior Diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction Diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.3.6. *Class Diagram* (Diagram Kelas)

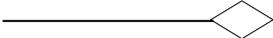
Menurut Sukamto et al (2014:141), “*Class Diagram* atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi”.

Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.1 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.	Kelas 	Kelas pada struktur sistem.
2.	Antarmuka/ <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi/ <i>Association</i> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	Asosiasi berarah/ <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .

Lanjutan **Tabel 2.1** Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus).
6.	Kebergantungan/ <i>dependency</i>	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.	Agregasi/ <i>aggregation</i> 	Relasi antarkelas dengan makna semua bagian (<i>whole-part</i>).

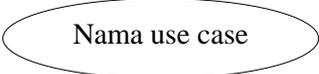
(Sumber: Sukamto et al, 2014:146)

2.3.7. Use Case Diagram

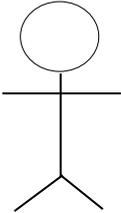
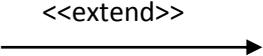
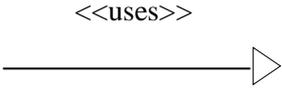
Menurut Sukamto et al (2014:155), “Use case atau diagram use case merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Secara kasar, *user case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu”.

Berikut adalah simbol-simbol yang ada pada diagram *use case*:

Tabel 2.2 Diagram *Use Case*

No	Simbol	Deskripsi
1.	<i>Use Case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar peran antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i> .

Lanjutan Tabel 2.2 Diagram *use case*

No	Simbol	Deskripsi
2.	Aktor/ <i>Actor</i> 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan kata benda di awal frase nama aktor.
3.	Asosiasi/ <i>association</i> 	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4.	Ekstensi/ <i>extend</i> 	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 itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek
5.	Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	Menggunakan/ <i>include/ uses</i> 	Relasi <i>use case</i> tambahan 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.

(Sumber: Sukanto et al, 2014:156)

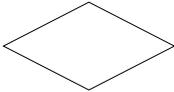
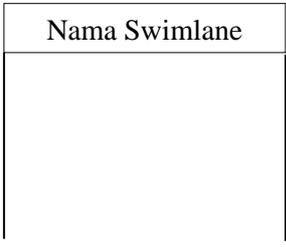


2.3.8. Activity Diagram (Diagram Aktivitas)

Menurut Sukamto et al (2014:161) mengenai pengertian *activity diagram*, “*Activity diagram* atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak.

Berikut simbol-simbol yang sering digunakan dalam *activity diagram*:

Tabel 2.3 Simbol-simbol *Activity Diagram*

No.	Simbol	Keterangan
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6.	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

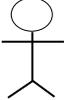
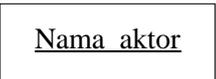
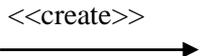
(Sumber: Sukamto et al, 2014:162)



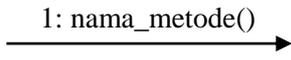
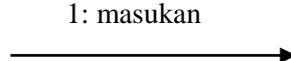
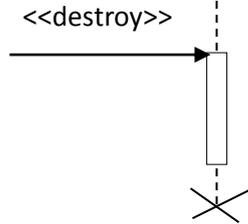
2.3.9. Sequence Diagram (Diagram Sekuensial)

Menurut Sukamto et al (2014:165), *Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

Tabel 2.4 Simbol-simbol *Sequence Diagram*

No.	Simbol	Keterangan
1.	<p>Aktor</p>  <p>Nama aktor Atau</p>  <p>Tanpa waktu aktif</p>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2.	<p>Garis hidup / <i>lifeline</i></p> 	Menyatakan kehidupan suatu objek.
3.	<p>Objek</p> 	Menyatakan objek yang berinteraksi dengan pesan.
4.	<p>Waktu aktif</p> 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
5.	<p>Pesan tipe <i>create</i></p> 	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.

Lanjutan **Tabel 2.4** Simbol-simbol *Sequence Diagram*

No.	Simbol	Keterangan
6.	Pesan tipe <i>call</i> 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek dikirim.
8.	Pesan tipe <i>return</i> 	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.
9.	Pesan tipe <i>destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

(Sumber: Sukamto et al, 2014:165)

2.3.10. Pengertian Kamus Data

Menurut Kristanto (2008:73), “Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu penggambaran atau pengidentifikasian setiap field atau file di dalam sistem”.

Menurut Sukamto (2013:73), “Kamus Data (*data dictionary*) adalah kumpulan daftar elemen yang mengalir pada sistem perangkat lunak sehingga



masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum atau memiliki standar penulisan”.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan, sebagai berikut:

Tabel 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Keterangan
1	=	Terdiri Atas
2	+	Dan
3	()	Opsional
4	[]	Memilih salah satu alternatif
5	**	Komentar
6	@	Identifikasi atribut kunci
7	L	Pemisah alternatif simbol []

(Sumber : Kristanto 2008:72)

2.3.11. Pengertian Basis Data (*Database*)

Menurut Sujatmiko (2012:76), “Basis data adalah kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan”.

Menurut Kristanto (2008:14), “Basis data adalah kumpulan data-data yang saling berhubungan dengan yang lain yang disimpan dalam perangkat keras komputer dan akan diolah menggunakan perangkat lunak”.

Menurut Priyadi (2014:2), “Secara umum, pengertian basis data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital.”

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. Tabel

Pada suatu basis data, tabel direpresentasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.



2. *Field*

Pada tabel basis data, kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertikal.

3. *Record*

Pada tabel basis data, baris merupakan suatu representasi untuk sebuah *record* yang pembacaan datanya dilakukan secara horizontal. Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu *record*. Nilai-nilai yang dimiliki oleh sebuah *record* merupakan gabungan dari semua *field* yang terdapat dalam tabel tersebut.

4. Kardinalitas

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya. Variasi kemungkinan untuk melakukan relasi yang dimiliki oleh kardinalitas terdiri dari empat macam, yaitu:

- a. Satu ke satu \longrightarrow (1:1).
- b. Satu ke banyak \longrightarrow (1:N).
- c. Banyak ke satu \longrightarrow (N:1).
- d. Banyak ke banyak \longrightarrow (N:N).

2.4. Teori Program

2.4.1. Pemrograman Visual Basic 2012

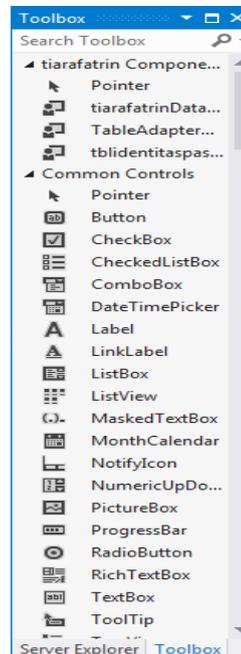
2.4.1.1. Pengertian Pemrograman *Visual Basic* 2012

Menurut Wahana Komputer (2013:2), “*Visual Basic* merupakan bahasa pemrograman yang paling populer dalam dunia komputer. Beraneka ragam program dapat dibuat dengan aplikasi ini”.

2.4.1.2 *Visual Studio Toolbox*

Menurut Wahana Komputer (2013:3), “*Toolbox* dapat diartikan sebagai kotak alat. Didalam *toolbox* terdapat banyak ikon di mana tiap ikon tersebut melambangkan objek yang disebut kontrol. Kontrol pada *toolbox* bisa digunakan

untuk membuat *instance* atau kejadian baru di *form* yang digunakan selama pemrograman”.



Gambar 2.3 *Toolbox All Windows Form*

(*Sumber: Wahana Komputer, 2013:4*)

Penjelasan fungsi-fungsi dari kontrol *toolbox* yang ada di *Visual Basic* 2012 seperti berikut:

Tabel 2.6 *Kontrol Toolbox Visual Basic 2012*

No	Nama	Keterangan
1.	<i>Pointer</i>	Untuk memilih <i>pointermouse</i> atau melepaskan dari memilih objek-objek <i>toolbox</i> .
2.	<i>Background Worker</i>	Mengeksekusi operator tertentu di <i>thread</i> lain.
3.	<i>Binding Navigator</i>	Menyediakan antarmuka untuk navigasi dan manipulasi data yang dikaitkan dengan <i>form</i> .
4.	<i>Binding Source</i>	Mengkapsulasikan sumber data untuk sebuah <i>form</i> dan menyediakan fasilitas <i>update</i> , <i>navigasi</i> , <i>filter</i> , dan <i>sorting</i> .

Lanjutan **Tabel 2.6** Kontrol *Toolbox Visual Basic* 2012

No.	Nama	Keterangan
5.	<i>Button</i>	Untuk menerima klik dari pengguna. Jika pengguna mengkliknya maka kode tertentu bisa dieksekusi.
6.	<i>Check Box</i>	Untuk menerima tanda centang dari pengguna. Jika pengguna mencentangnya maka item di kotak tersebut otomatis bisa dipakai untuk program.
7.	<i>Checked List Box</i>	Menampilkan item-item seperti <i>listbox</i> biasa, namun dengan <i>checkbox</i> disebelah kirinya sehingga satu <i>listbox</i> bisa menerima dua item atau lebih.
8.	<i>Combo Box</i>	Menampilkan item-item namun dalam bentuk hanya satu yang terlihat. Jika pengguna mengklik <i>combobox</i> barulah semua item didalamnya akan terlihat.
9.	<i>Context Menu Strip</i>	Menampilkan menu <i>shortcut</i> jika pengguna mengklik kanan pada item.
10.	<i>Data Grid View</i>	Menampilkan data dalam bentuk table yang bisa dimodifikasi.
11.	<i>Data Set</i>	Merupakan representasi data dari sumber data di memori.
12.	<i>Date Time Picker</i>	Memungkinkan pengguna menampilkan tanggal dan memilihnya. Selanjutnya tanggal dan waktu dipilih dapat dimodifikasi menggunakan format tertentu.
13.	<i>Directory Entry</i>	Mengenkapsulasi node atau objek di hirarki <i>Active Directory (AD)</i> di <i>Windows Server</i> .
14.	<i>Directory Searcher</i>	Melakukan <i>query</i> terhadap <i>active directory</i> .
15.	<i>Domain Up Down</i>	Menampilkan nilai <i>string</i> yang ditampilkan pengguna dari daftar item dengan cara mengklik tombol up dan down.
16.	<i>Error Provider</i>	Menentukan dukungan untuk inisialisasi transaksi.
17.	<i>Event Log</i>	Menyediakan interaksi dengan <i>eventlog</i> dari windows.
18.	<i>File System Watcher</i>	Memonitor perubahan sistem file dan mengeksekusi event ketika ada perubahan direktori atau file.

Lanjutan Tabel 2.6 Kontrol *Toolbox Visual Basic 2012*

No.	Nama	Keterangan
19.	<i>Group Box</i>	Mengelompokkan control dalam satu kotak untuk mempermudah pengaturan dan manajemen antarmuka.
20.	<i>Image List</i>	Menampilkan daftar gambar.
21.	<i>Label</i>	Untuk menampilkan teks dimana nantinya isi didalamnya bisa diberi masukan oleh pengguna seperti halnya <i>textbox</i> .
22.	<i>List Box</i>	Menampilkan item-item dalam bentuk kotak di mana semua item ditampilkan di dalam daftar.
23.	<i>List View</i>	Menampilkan item-item dalam satu dari lima tampilan yang berbeda.
24.	<i>Radio Button</i>	Memungkinkan pengguna untuk memilih satu opsi dari beberapa pilihan yang dikaitkan dengan tombol radio lainnya.
25.	<i>Text Box</i>	Memungkinkan pengguna memasukan teks yang bisa lebih dari satu baris dan menyediakan kemampuan pengeditan dan <i>masking</i> kata sandi.

(Sumber: Wahana Komputer, 2013:4)

2.4.1.3 Panel *Properties*

Panel *properties* menampilkan *properties* dari kontrol yang dipilih. Berikut ini merupakan contoh tampilan jendela *properties* untuk kontrol *form* dan arti-artinya dari nilai *properties*-nya:


Gambar 2.4 Panel *properties*

(Sumber: Wahana Komputer, 2013: 16)

Tabel 2.7 Kontrol *Properties Visual Basic 2012*

No.	Nama	Keterangan
1.	<i>Auto Scroll</i>	Mengindikasikan apakah <i>scrollbar</i> otomatis diaktifkan jika isi kontrol lebih besar dari area yang terlihat.
2.	<i>Auto Size</i>	Menentukan apakah kontrol akan mengatur ulang ukuran dirinya sendiri secara otomatis sesuai dengan kontennya.
3.	<i>Back Color</i>	Menentukan warna <i>background</i> dari komponen.
4.	<i>Background Image</i>	Menentukan gambar <i>background</i> untuk kontrol.
5.	<i>Control Box</i>	Menentukan apakah <i>form</i> memiliki kontrol/ sistem <i>menubox</i> atau tidak.
6.	<i>Enabled</i>	Menentukan apakah kontrol dalam posisi aktif atau tidak.
7.	<i>Font</i>	Menentukan jenis dan ukuran <i>font</i> .
8.	<i>Fore Color</i>	Menentukan warna <i>foreground</i> / depan dari komponen yang digunakan untuk menampilkan teks dikomponen tersebut.
9.	<i>Icon</i>	Mengindikasikan <i>icon</i> untuk sebuah <i>form</i> . <i>Icon</i> ini akan ditampilkan dibagian <i>menu box</i> sistem dan ketika <i>form</i> sedang di <i>minimize</i> .
10.	<i>Locked</i>	Menentukan apakah kontrol bisa dipindahkan atau di <i>resize</i> .
11.	<i>Show Icon</i>	Menentukan apakah <i>icon</i> ditampilkan di <i>titlebar</i> di <i>form</i> .
12.	<i>Maximize Box</i>	Menentukan apakah <i>form</i> memiliki tombol <i>maximize</i> dibagian kanan atas di <i>captionbar</i> .



Lanjutan Tabel 2.7 Kontrol *Properties Visual Basic* 2012

No.	Nama	Keterangan
13.	<i>Maximum Size</i>	Menentukan ukuran maksimal <i>form</i> yang bisa di <i>resize</i> .
14.	<i>Minimize Box</i>	Menentukan apakah <i>form</i> memiliki kotak <i>minimize</i> dibagian kanan atas dari <i>captionbar</i> .
15.	<i>Minimum Size</i>	Menentukan ukuran minimal dari <i>form</i> yang bisa di <i>resize</i> .
16.	<i>Opacity</i>	Menentukan persen ketidaktransparan dari kontrol, 0% artinya transparan dan 100% tidak transparan.
17.	<i>Size</i>	Menentukan ukuran control.
18.	<i>Text</i>	Menentukan teks yang diasosiasikan untuk kontrol tersebut.
19.	<i>Window State</i>	Menentukan kondisi awal jendela dari <i>form</i> .

(*Sumber: Wahana Komputer, 2013:17*)

2.4.2 *SQL Server 2008*

2.4.2.1 Pengertian *SQL Server 2008*

Menurut Wahana Komputer (2010:2), *SQL Server 2008* adalah sebuah terobosan baru di Microsoft dalam bidang *database*. *SQL Server* adalah sebuah DBMS (*Database Management System*) yang dibuat oleh Microsoft untuk ikut berkecimpung dalam persaingan dunia pengolahan data menyusul pendahulunya seperti IBM dan Oracle. *SQL Server 2008* dibuat pada saat kemajuan dalam bidang *hardware* sedemikian pesat. Oleh karena itu sudah dapat dipastikan bahwa *SQL Server 2008* membawa beberapa terobosan dalam bidang pengolahan dan penyimpanan data.

2.4.2.2 Kebutuhan *Hardware SQL Server 2008*

Menurut Wahana Komputer (2010:2), Sebelum melakukan proses *download* dan instalasi *SQL Server 2008*, pahamiilah terlebih dahulu *hardware* yang harus



digunakan untuk menjalankan *SQL Server*. Adapun *hardware* yang diperlukan minimal adalah sebagai berikut:

1. *Processor* \geq 1 GHz.
2. *Memory* \geq 512 Mb.
3. OS = Windows XP 32 bit, Windows 7.

Biarapun dapat diinstal pada sistem komputer dengan memori 512 MB, tetapi disarankan menggunakan memori 1 GB. Sedangkan untuk jaringannya diperlukan adalah sebagai berikut:

1. *Shared Memory*
2. TCP/ IP
3. *Named Pipes*
4. VIA (*Virtual Interface Adapter*)

Sedangkan bila menggunakan Windows XP Profesional maka pastikan bahwa pada sistem anda telah terdapat .Net Framework 3.5 SP 1 dan juga *update*-nya, kemudian Windows Installer 4.5 dan Windows Power Shell.

2.4.2.3 Kunci dalam *SQL Server 2008*

Wahana Komputer (2013:220), Kunci primer merupakan atribut (satu kolom) atau kombinasi kolom yang nilainya unik (tidak ganda) dimana dengan kunci tersebut dapat memberi identitas pada sebuah *record* dalam satu entitas.

Ketika memilih kunci untuk entitas, sebuah kolom haruslah memiliki beberapa karakteristik:

1. Tiap *record* di entitas tersebut tidak boleh memilih nilai kosong (*null*) atau dengan kata lain nilainya harus ada (*not null*).
2. Nilainya harus unik dan tidak ada yang ganda.
3. Nilainya tidak berubah atau menjadi kosong ketika diperlukan.
4. Hanya ada satu kunci primer yang didefinisikan untuk tiap entitas.

Kadang diperlukan kombinasi lebih dari satu kolom sebagai identitas unik dari table. Untuk menerangkan istilah kunci primer yang terdiri atas lebih dari satu *field* ini disebut kunci komposit (*composite key*).



Foreignkey atau kunci asing merupakan atribut yang akan melengkapi relasi dengan cara mengidentifikasi entitas induknya. Kunci asing memiliki metode yang berguna untuk menjaga integritas data dan untuk melakukan navigasi antara beberapa table. Setiap relasi di table harus didukung oleh kunci asing.