

BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Perancangan

Tahap dimana analisis sistem membentuk sistem disebut dengan perancangan sistem. Kegiatan perancangan sistem dikerjakan setelah tahap analisis, karena setelah kegiatan analisis sistem seorang analis sistem telah mendapatkan gambaran dengan jelas apa yang dikenal harus dikerjakan. Tahapan perancangan sistem bisa dibagi menjadi dua bagian^[1].

1. Merupakan tahap lanjutan setelah tahap analisis sistem dalam daur hidup pengembangan sistem
2. Mendefinisikan setiap kebutuhan-kebutuhan fungsional.
3. Mempersiapkan rancangan implementasi sistem yang baru/usulan.
4. Menggambarkan sistem baru/usulan yang akan dikembangkan.
5. Mengatur dan merencanakan elemen-elemen yang terpisah serta mengkonfigurasi perangkat lunak dan keras.

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tentu yaitu sebagai berikut :

a. Komponen Sistem (*components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.

b. Batas Sistem (*boundary*)

Merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

c. Lingkungan Luar Sistem (*environment*)

Adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem.

d. Penghubungan Sistem (*interface*)

Merupakan media penghubung antara satu subsistem dengan subsistem lainnya.

e. Masukan Sistem (*input*)

Adalah energy yang dimasukan ke dalam sistem.

f. Keluaran Sistem (*output*)

Adalah hasil dari energy yang di olah dan diklasifikasikan menjadi keluaran yang berguna.

g. Pengelolaan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran.

h. Sasaran Sistem

Suatu sistem dapat mempunyai tujuan (*goal*) atau sasaran (*objective*)

2.1.2 Pengertian Informasi

Informasi adalah data yang telah diklasifikasi atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.^[02]

2.1.3 Pengertian Sistem Informasi

Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manjerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.^[03]

Sistem Informasi merupakan suatu komponen yang saling berhubungan dengan proses penciptaan dan penyampaian informasi dalam perusahaan, yang memproses input berupa sumber data, kemudian diproses dengan komponen hardware, software, dan brainware dan menghasilkan informasi sebagai output.^[03]

2.2 Kepegawaian

2.2.1 Pengawai

Menurut Kamus Besar Indonesia. Kepegawaian adalah yang berhubungan dengan pengawai atau orang yang bekerja pada pemerintah dan perusahaan. [04]

2.2.2 Jabatan (JOB)

Merupakan sekumpulan Pekerjaan (JOB) yang berisi tugas-tugas yang sama atau berhubungan satu dengan yang lain.

2.2.3 Mengajar

Menurut S. Nasution (1985:26), Mengajar adalah suatu aktivitas mengorganisasi lingkungan sebaik-baiknya dan menghubungkan dengan siswa sehingga terjadi proses belajar.[06]

2.2.4 Pendidikan

Menurut UU Nomor 2 Tahun 1989 Pendidikan adalah usaha sadar untuk menyiapkan peserta didik melalui kegiatan bimbingan, pengajaran dan atau latihan bagi peranannya di masa yang akan datang.[07]

2.3 Sistem Perangkat Lunak yang digunakan

2.3.1 Pengertian *Visual Studio 2019*

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, Integrated Development Environment (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual

SourceSafe.

Microsoft Visual Studio memiliki tiga edisi yakni:

a. Edisi Komunitas

Ini adalah versi gratis yang diumumkan pada 2014, adapun edisi lainnya berbayar. Ini berisi fitur yang mirip dengan edisi Profesional. Menggunakan edisi ini, setiap pengembang dapat mengembangkan aplikasi gratis atau berbayar mereka sendiri seperti .Net aplikasi, aplikasi Web dan banyak lagi.

b. Edisi Profesional

Ini adalah edisi komersial Visual Studio. Muncul dalam Visual Studio 2010 dan versi yang lebih baru. Pada edisi ini memberikan dukungan untuk mengedit XML dan XSLT dan termasuk alat seperti Server Explorer dan integrasi dengan Microsoft SQL Server. Microsoft menyediakan uji coba gratis untuk edisi ini dan setelah periode uji coba, pengguna harus membayar untuk terus menggunakannya. Tujuan utamanya adalah untuk menyediakan Fleksibilitas (alat pengembang profesional untuk membangun semua jenis aplikasi), Produktivitas (Fitur hebat seperti CodeLens), Kolaborasi (alat perencanaan proyek Agile, grafik, dll.) Dan manfaat Pelanggan seperti perangkat lunak Microsoft, plus Azure, Pluralsight, dan lain – lain.

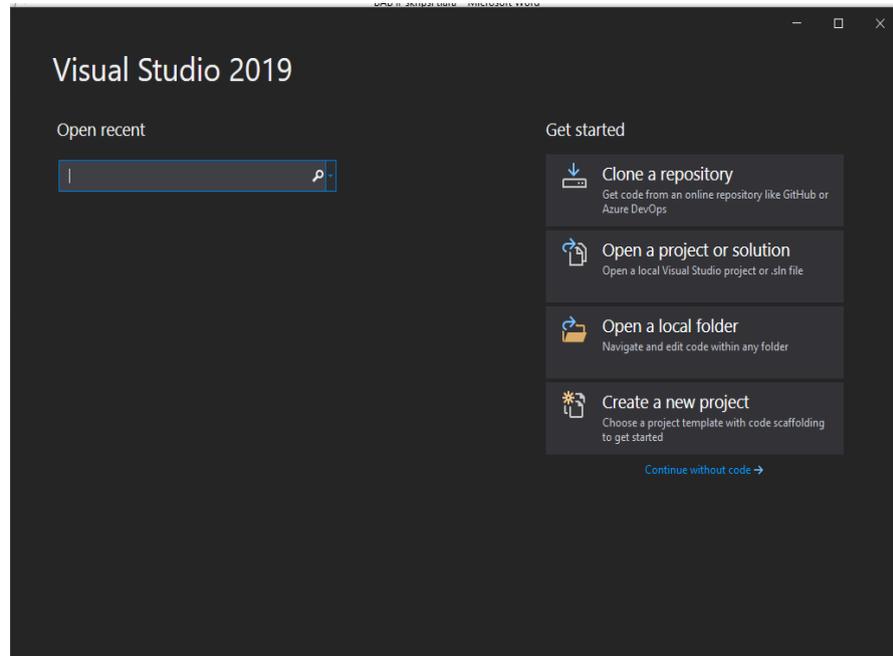
c. Edisi Perusahaan

Ini adalah solusi ujung ke ujung yang terintegrasi untuk tim dengan ukuran berapa pun dengan kualitas dan kebutuhan skala yang menuntut. Microsoft menyediakan uji coba gratis 90 hari untuk edisi ini dan setelah periode uji coba, pengguna harus membayar untuk terus menggunakannya. Manfaat utama dari edisi ini adalah sangat dapat diskalakan dan memberikan perangkat lunak berkualitas tinggi.

Berikut cara memulai program *Visual Studio 2019*:

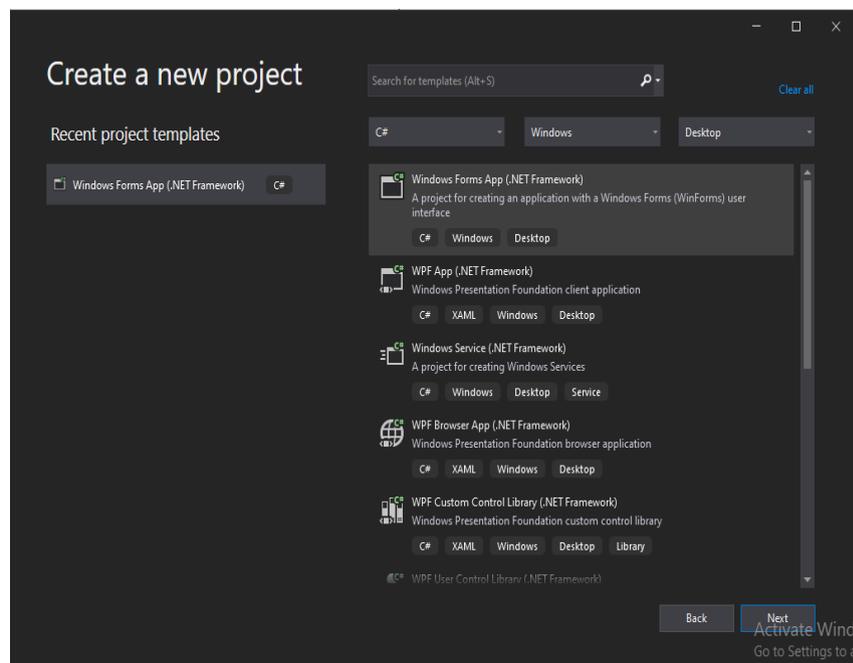
a. Klik *icon* start pada Sistem Operasi Windows

b. Klik Program, lalu Pilih *icon visual studio 2019*, maka akan tampil seperti gambar dibawah ini :



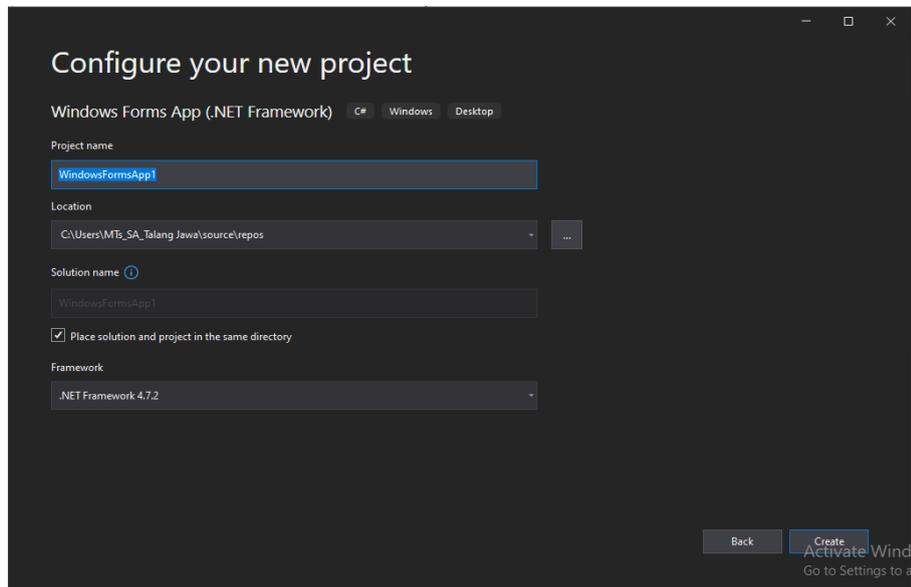
Gambar 2.1. Tampilan Dasar *Visual Studio 2019*

c. Pilih *Create a new project*, maka akan tampil gambar dibawah ini :



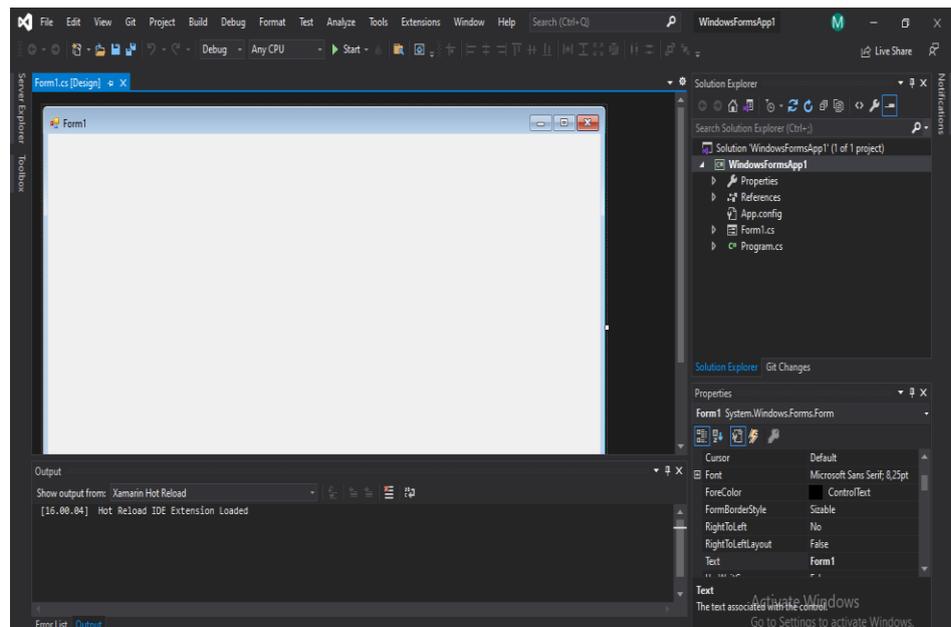
Gambar 2.2. Tampilan *Create a New Project Visual Studio 2019*

- d. Pilih menu *windows form app (.Net Fraework) C#* , kemudian *klik next*, maka akan tampil gambar dibawah ini:



Gambar 2.3. Tampilan *Configure your New Project*

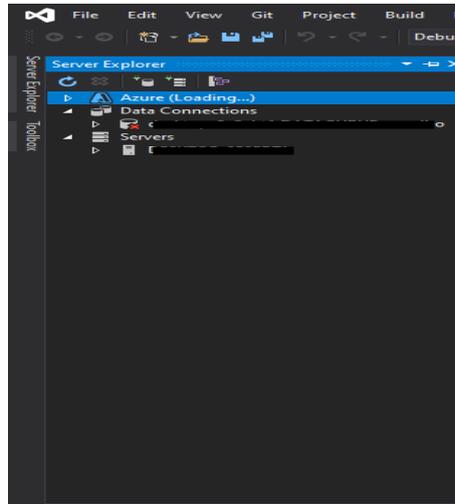
- e. Isilah nama project sesuai dengan apa yang diinginkan, pada *coloumn Project name*. Kemudian *klik create*, maka akan tampil gambar dibawah ini :



Gambar 2.4. Tampilan Area Kerja *C# Visual Studio 2019*

f. *Server Explorer*

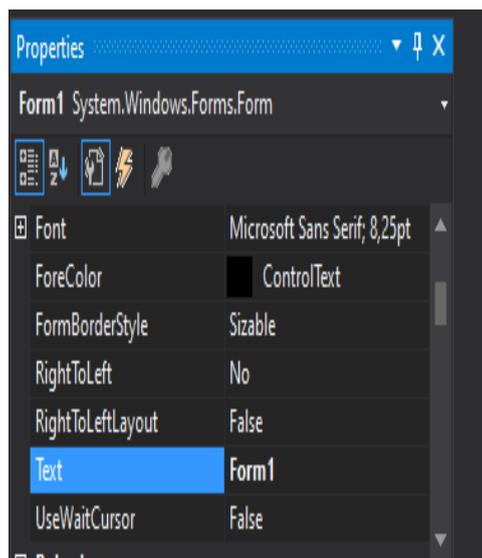
Berfungsi untuk antarmuka koneksi antara *form aplikasi* ke *server data base*.



Gambar 2.5 Jendela Server Explore

g. *Properties*

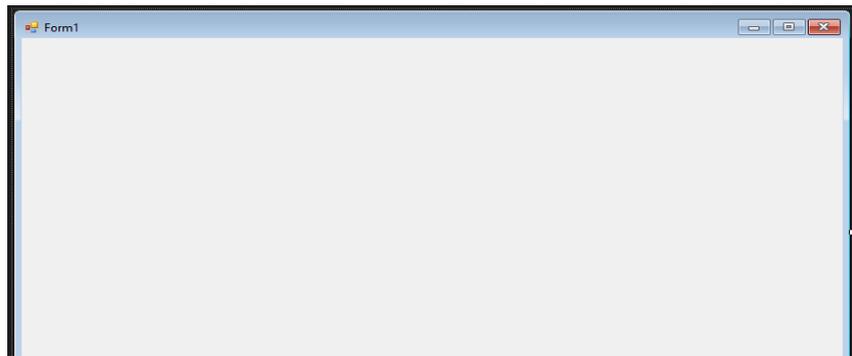
Berfungsi untuk mengubah sifat-sifat yang dibawa oleh *object* tersebut.



Gambar 2.6 Jendela *Properties*

h. *Form Designer*

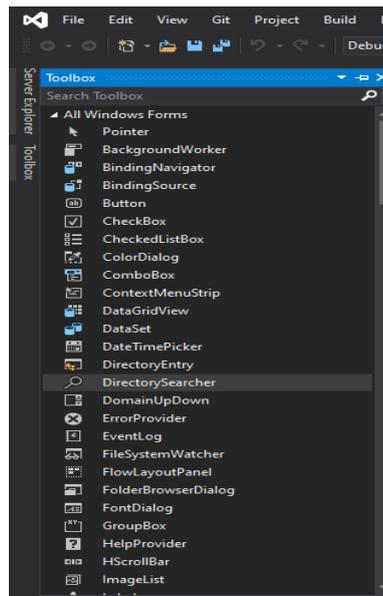
Form Designer berfungsi untuk mendesain tampilan aplikasi. Dan tempat meletakkan komponen-komponen dari *tool palette* ke dalam *form*. Setelah meletakkan komponen tersebut kita dapat mengubah tampilan sesuai dengan aplikasi yang kita buat.



Gambar 2.7 *Form Designer*

i. *Component Tool Box*

Component Tool Box berisi daftar komponen yang disediakan oleh C# di *visual studio 2019*. Komponen-komponen ini di sajikan dalam halaman-halaman tab dan dikelompokkan berdasarkan jenisnya.



Gambar 2.8 *Jendela Komponen Tool Box*

2.3.2 Pengertian *Microsoft SQL Server Management Studio*

Microsoft SQL Server Management Studio yang biasa disingkat SSMS merupakan sebuah produk RDBMS (Relational Database Management System) dari *Microsoft SQL Server Management Studio* digunakan untuk mengkonfigurasi, dan mengelola komponen yang ada didalam *Microsoft SQL Server Management Studio*. *Microsoft SQL Server Management Studio* banyak digunakan pada bidang bisnis, pendidikan, maupun pemerintahan sebagai solusi untuk penyimpanan data. Kelebihan *Microsoft SQL Server Management Studio* salah satunya yaitu mudah dalam melakukan koneksi dengan komputer client yang pembangunan aplikasinya menggunakan software yang sama platform dengan *Microsoft SQL*, misalnya *Microsoft Visual Studio*. Perintah dasar yang terdapat pada *SQL (Structured Query Language)* adalah sebagai berikut:

1. Membuat Database

```
CREATE DATABASE nama_database
```

2. Membuat Tabel

```
CREATE nama_tabel (nama_kolom tipe_data)
```

3. Memasukkan Data

```
INSERT INTO nama_tabel (nama_kolom) values ('isi_kolom')
```

4. Menghapus Data

```
DELETE FROM nama_tabel where condition
```

5. Memilih Data

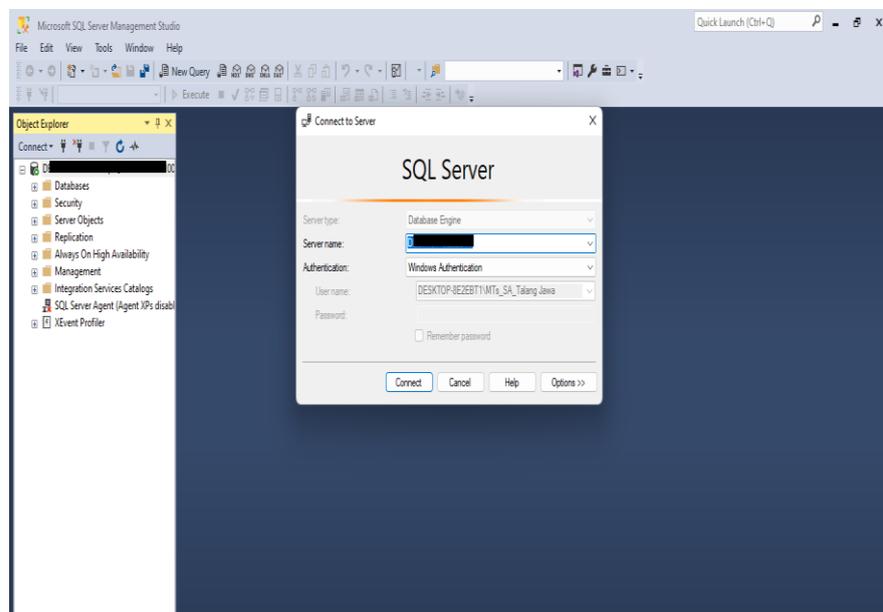
```
SELECT * FROM nama_tabel
```

6. Mengubah Data

```
UPDATE nama_tabel SET nama_kolom='values'
```

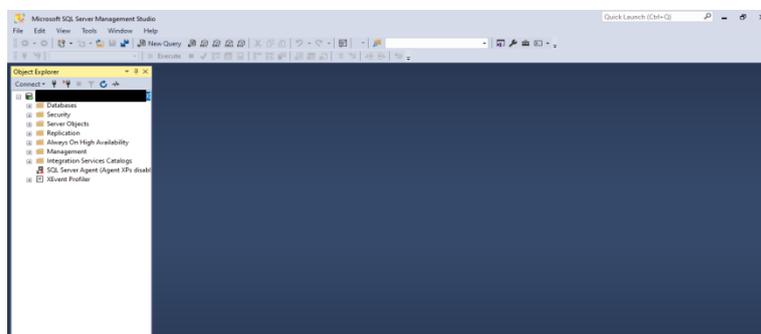
Adapun untuk memulai program *Microsoft SQL Serve Management Studio* adalah sebagai berikut

- Klik icon Start pada Sistem Operasi *Windows*.
- Klik Pilihan *all program*, kemudian pilih *SQL Server*, maka akan tampil gambar seperti dibawah ini:



Gambar 2.9 Jendela SSMS

- Klik pilih *Connect*, maka akan tampil gambar seperti di bawah ini :



Gambar 2.10 Area Kerja SSMS

1) Objek *Microsoft SSMS*

a) *Table*

Table Adalah sekumpulan data atau informasi speksifik tentang subjek tertentu yang disusun dalam bentuk kolom dan baris. Kolom atau yang lebih dikenal dengan istilah *field* berisi kumpulan data yang memiliki karakteristik pengertian yang sama.

b) *Query*

Query berarti “Mendefinisikan Data” yakni memanipulasi data dan mengendalikan memanipulasi data tersebut melalui bahasa Selanjutnya, bahasa yang digunakan yaitu bahasa *Query*.

c) *Form*

Form dirancang untuk menampilkan *field-field* yang dibutuhkan dan label dalam penjelasannya dalam format tampilan yang lebih menarik.

d) *Report*

Report (Laporan) digunakan untuk menampilkan dan untuk mencetak informasi yang berasal dari table atau *query*.

e) *Pages*

Pages adalah sebuah objek khusus yang digunakan untuk menampilkan dan bekerja dengan data yang diambil dari internet.

f) *Macro*

Macro adalah rangkaian dari beberapa perintah yang dapat disimpan dan dijalankan ulang secara otomatis, misalnya membuka *form* mencetak *report* dan sebagainya.

2) Tipe Data

a) *Text*

Serangkaian huruf, angka, spasi, dan tanda baca.

b) *Memo*

Serangkaian kata yang sangat panjang berupa data campuran antara huruf, spasi dan tanda baca

c) *Number*

Data berupa angka untuk perhitungan numeris.

d) *Data Time*

Data yang berupa tanggal atau waktu

e) *Currency*

Data yang berupa nilai mata uang untuk perhitungan numeris

f) *Auto Number*

Bilangan atau penomoran yang berurutan secara otomatis.

g) *Yes/No*

Data pemasukan hanya berupa penyangkalan antara *yes* atau *no*

h) *OLE Object*

OLE Object Embedded (OLE) object merupakan tipe data yang berupa objek atau gambar.

i) *Hyperlink*

Data yang dapat dihubungkan atau dikaitkan dengan alamat *hyperlink* pada lokasi penyimpan

j) *Lookup wizard*

Membuat data dengan memilih salah satu data pada daftar table.

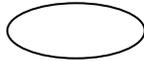
2.3.3 Pengertian DFD (*Data Flow Diagram*)

Data Flow Diagram sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir (misalnya lewat telpon, surat dan sebagainya) atau lingkungan fisik dimana data tersebut akan disimpan (misalnya file kartu, *microfiche*, *hard disk*, *tape*, *diskette*, dan lain sebagainya).^[11]

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur (*structured Analysis and design*). DFD merupakan alat yang cukup populer sekarang ini, karena dapat menggambarkan arus data didalam sistem dengan terstruktur dan jelas. Lebih lanjut DFD juga merupakan dokumentasi dari sistem yang baik.

2.3.4 Diagram Konteks

- a. Diagram Konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau *output* dari sistem. Dalam diagram konteks biasanya hanya ada satu proses. Tidak boleh ada *store* dalam diagram konteks.

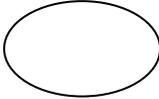
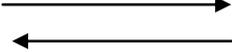
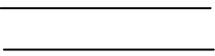
Simbol	Arti
	Menunjukkan sebuah sistem
	Menunjukkan bagian
	Menunjukkan alir data

Tabel 2.1. Simbol-simbol di diagram konteks

- b. Diagram Rinci (*Level Diagram*)

Level Diagram adalah diagram yang menguraikan proses apa yang ada dalam diagram level di atasnya. DFD Level terdiri dari DFD level 0, DFD level 1, DFD level 2 dan seterusnya

Simbol	Keterangan
	Entitas Entitas (External Entity merupakan lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada dilingkungan luarnya yang akan memberikan input atau menerima output dari sistem.
	<u>Proses</u>

	<p>Proses adalah kerja yang dilakukan oleh orang atau computer dari hasil suatu aliran data yang masuk ke dalam proses untuk dihasilkan aliran data yang akan keluar dari proses</p>
	<p>Aliran data Aliran data mengalir diantara proses (<i>process</i>), simpanan data (<i>data store</i>) dan kesatuan luar (<i>entity</i>). Aliran data ini menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari sistem.</p>
	<p>Penyimpanan Data (<i>Data Store</i>) Data store menunjukkan penyimpanan file atau database</p>

Tabel 2.2 Simbol-simbol *Data Flow Diagram*

2.3.5 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) adalah alat permodelan data utama dan akan membantu mengorganisasi data dalam satu proyek kedalam entitas-entitas dan menentukan hubungan antar entitas. Proses memungkinkan analisis menghasilkan struktur basis data yang baik sehingga data dapat disimpan dan diambil secara efisien

a. Entitas

Adalah suatu data yang dapat disimpan dan berguna bagi badan atau perusahaan, dengan kata lain objek yang diidentifikasi atau dibedakan dengan objek lainnya. Digambarkan dengan kotak persegi panjang.



Gambar 2.11 Lambang Entitas

b. Atribut

Atribut menunjukkan karakteristik dari tiap-tiap entitas. Atribut

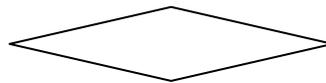
digambarkan dengan bentuk oval.



Gambar 2.12 Lambang Atribut

c. *Relasi*

Relasi menunjukkan hubungan yang terjadi antara entitas. Relasi digambarkan dalam bentuk belah ketupat atau *diamond*



Gambar 2.13 Lambang Relasi

d. *Line Connector*

Line Connector digambarkan dengan bentuk garis tunggal

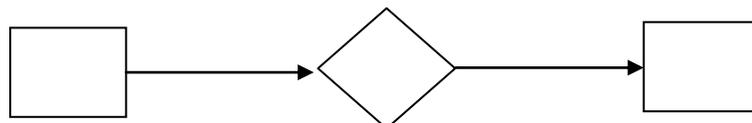


Gambar 2.14 Lambang *Line Connector*

Tabel-tabel dapat direlasikan satu sama lain dan dapat diakses seakan-akan mereka adalah satu gabungan tabel yang terdiri atas banyak kolom dan baris. Jumlah entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain tersebut kardinalitas derajat relasi. Terdapat tiga macam kardinalitas relasi yaitu :

1) Hubungan Satu pada Satu (*One to One* atau 1:1)

Hubungan satu ke satu dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan dengan satu kejadian pada entitas yang kedua dan sebaliknya.

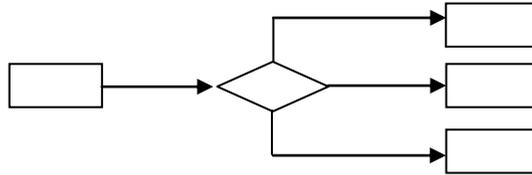


Gambar 2.15 Hubungan satu pada satu

2) Hubungan Satu pada banyak (*One to Many* atau 1:M)

Tingkat hubungan satu ke banyak adalah sama dengan banyak ke satu.

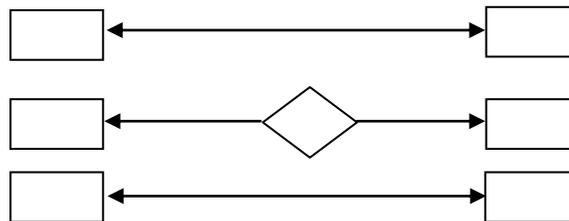
Untuk suatu kejadian pada entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua.



Gambar 2.16 Hubungan satu pada banyak

3) Hubungan Banyak pada Banyak (*Many to Many atau M:M*)

Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas lainnya



Gambar 2.17 Hubungan banyak pada banyak