



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Hartono (2013:27) berpendapat bahwa, “Komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang untuk secara otomatis melakukan serangkaian urutan penghitungan (*arithmetic*) atau proses–proses yang diurutkan secara logis”.

Sujatmiko (2013:156) mengemukakan “Komputer adalah mesin yang dapat mengelola data digital dengan serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi informasi”.

Ariyus dan Pangera (2010:3) menyatakan “Komputer adalah perangkat elektronik, beroperasi di bawah perintah pengendali yang disimpan pada memori komputer, dapat menerima data, memproses data berdasarkan aturan tertentu, mencetak hasilnya, dan menyimpan data untuk penggunaan di masa depan”.

2.1.2. Pengertian Data

Ladjamudin (2013:8) mendefinisikan “Data adalah deskripsi dari sesuatu dan kejadian yang kita hadapi (*the description of thing and events that we face*). Definisi data yang lain adalah data merupakan kenyataan yang menggambarkan suatu kejadian dan kesatuan nyata”.

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

Kristanto (2008:7) mendefinisikan data sebagai “Penggambaran dari suatu kejadian yang sedang terjadi, dimana data tersebut akan diolah dan diterapkan dalam sistem menjadi input yang berguna dalam suatu sistem”.

Selain itu, Dhanta (2009:212) menyatakan “Data adalah sesuatu yang dapat dilambangkan, dikodekan, atau didigitalisasi ke dalam lambang–lambang atau kode–kode yang dimengerti oleh komputer”.



2.1.3. Pengertian Perangkat Lunak (*Software*)

Ladjamudin (2013:20) mendefinisikan “*Software* merupakan kumpulan dari perintah/fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu”.

Rosa (2013:2) menyatakan “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).”

Sujatmiko (2012:201) berpendapat “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”.

2.1.4. Pengertian Basis Data (*Database*)

Sujatmiko (2012:40) mengemukakan “Basis data (*database*) merupakan kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut”.

Ladjamudin (2013:21) berpendapat “*Database* merupakan kumpulan file yang saling terintegritasi, namun *database* tidak akan dapat diakses oleh siapapun tanpa adanya *software* aplikasi”.

Selain itu, Sukanto dan Shalahuddin (2014:43) menyatakan, “Basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

2.1.5. Pengertian Aplikasi

Sujatmiko (2012:23) menyatakan “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas–tugas tertentu, misalnya MS-Word, MS-Excel”.



2.1.6. Pengertian Informasi

Hartono (2013:15) menyatakan, “Informasi pada dasarnya adalah sehimpunan data yang telah diolah menjadi sesuatu yang memiliki arti dan kegunaan lebih luas”.

Gordon.B.Davis (1995) dalam Ladjamudin (2013:8) mendefinisikan “Informasi sebagai data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang”.

Sujatmiko (2012:135) menyatakan, “Informasi merupakan serangkaian simbol yang jika disusun menurut aturan tertentu dapat menyampaikan suatu pesan”.

2.1.7. Pengertian Internet

Sujatmiko (2012:138) berpendapat bahwa, “Internet atau *Interconnected Network* adalah jaringan global yang menghubungkan berjuta-juta komputer di seluruh dunia melalui jalur telepon kabel maupun satelit. Dengan adanya internet, jutaan orang di seluruh duniaberbagi segala jenis informasi”.

2.2. Teori Khusus

2.2.1. Data Flow Diagram (DFD)

Kristanto (2008:55) menyatakan, “*Data flow diagram*/DFD adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana 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”.

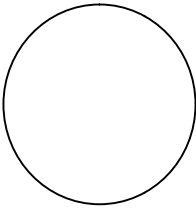
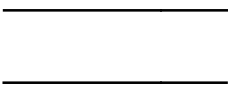
Selain itu, Sukamto dan Shalahuddin (2013:70) menyatakan, “*Data Flow Diagram* atau dalam bahasa Indonesia menjadi diagram alir data adalah representatif grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).



Adapun simbol–simbol atau notasi–notasi yang menggambarkan *Data Flow Diagram* (DFD), sebagai berikut:

1. Teknik Edward Yourdon dan Tom DeMarco

Tabel 2.1. Simbol–simbol *Data Flow Diagram* (DFD) menurut Edward Yourdon dan Tom DeMarco

Notasi	Keterangan
	Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harus menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.
	File atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel–tabel basis data yang dibutuhkan, tabel–tabel ini juga harus sesuai dengan perancangan tabel–tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM)). Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya kata benda.



Notasi	Keterangan
	Entitas luar (<i>External Entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan. Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.
	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan: Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.

(Sumber: Sukamto dan Shalahuddin, 2014:71-72)

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membentuk DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.



2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul–modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul–modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 dan seterusnya.

4. Membuat DFD Level 3 dan seterusnya

DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

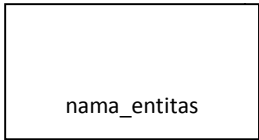
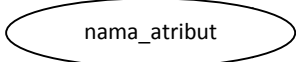
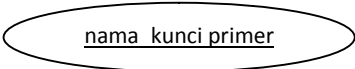
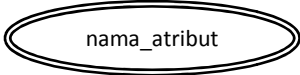
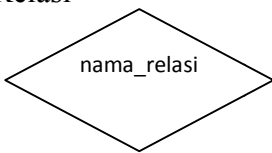
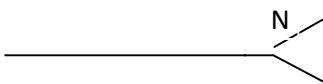
2.2.2. Pengertian *Entity Relationship Diagram* (ERD)

Pratama (2014:49) mendefinisikan “ERD (*Entity Relationship Diagram*) adalah diagram yang menggambarkan keterkaitan antartabel beserta *field-field* di dalamnya pada suatu database sistem.

Ladjamudin (2013:142) menyatakan, “*Entity Relationship Diagram (ERD)* adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

Adapun simbol-simbol yang digunakan dalam *Entity Relationship Diagram* (ERD), yaitu:

**Tabel 2.3.** Simbol–Simbol *Entity Relationship Diagram* (ERD)

Simbol	Deskripsi
Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
AtributKunciPrimer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
AtributMultinilai/ <i>Multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
Asosiasi/ <i>Association</i> 	Penghubung antara relasi dan entitas di mana dikedua ujungnya punya <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

(Sumber: Sukamto dan Shalahuddin, 2013:50-51)



2.2.3. Pengertian *Flow Chart*

Ladjamudin (2013:263) menyatakan, “*Flowchart* adalah bagan–bagan yang mempunyai arus yang menggambarkan langkah–langkah penyelesaian suatu masalah”.

2.2.3.1. Simbol-simbol Flowchart

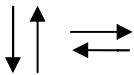
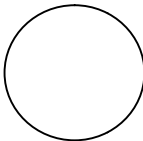
Flowchart disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses didalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut.

1. *Flow Direction Symbols* (Simbol penghubung/alur)
2. *Processing Symbols* (Simbol proses).
3. *Input-output Symbols* (Simbol input-output)

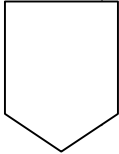
2.2.3.1.1. Flow Direction Symbols

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut.

Tabel 2.4. Flow Direction Symbols

No.	Simbol	Arti
1.		Simbol arus / flow Untuk menyatakan jalannya arus suatu proses.
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya.
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.



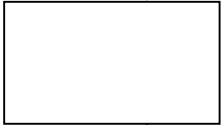
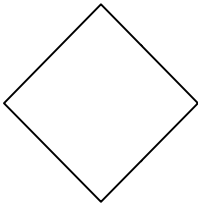
No.	Simbol	Arti
4.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.

(Sumber: Ladjamudin, 2013:266)

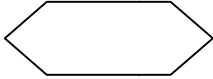
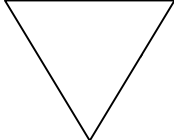
2.2.3.1.2. Processing Symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses/prosedur. Simbol-simbol tersebut adalah sebagai berikut.

Tabel 2.5. Processing Symbols

No.	Simbol	Arti
1.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.
2.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
3.		Simbol Decision/logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/ tidak.



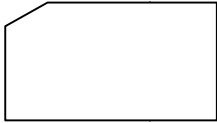
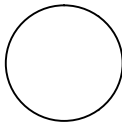
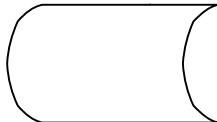
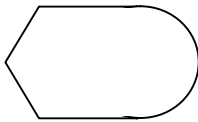
No.	Simbol	Arti
4.		Simbol Predefined Proses Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program.
6.		Simbol Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		Simbol Offline Storage Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.		Simbol Manual Input Untuk memasukkan data secara manual dengan menggunakan online keyboard.

(Sumber: Ladjamudin, 2013:267)

2.2.3.1.3. Input-output Symbols

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media input atau output. Simbol-simbol tersebut adalah sebagai berikut.

**Tabel 2.6.** Input-output Symbols

No.	Simbol	Arti
1.		Simbol Input-output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
2.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output dituliskan ke kartu.
3.		Simbol Magnetic-tape Unit Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic.
4.		Simbol Disk Storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
5.		Simbol Document Untuk mencetak laporan ke printer.
6.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

(Sumber: Ladjamudin, 2013:268)

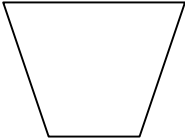
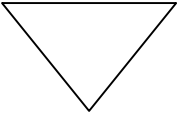
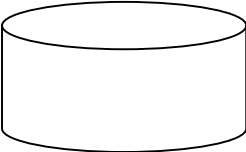


2.2.4. Pengertian *Block Chart*

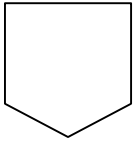
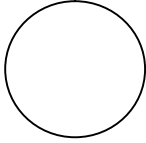
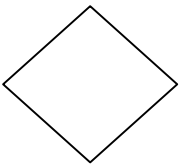
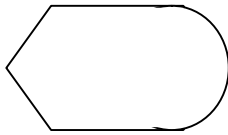
Kristanto(2008:68), “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol–simbol tertentu”. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Adapun simbol–simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.7. Simbol-simbol dalam *Blockchart*

Simbol	Keterangan
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
	Multi dokumen
	Proses manual
	Proses yang dilakukan oleh komputer
	Menandakan dokumen yang diarsipkan (arsip manual)
	Data penyimpanan (data storage)



Simbol	Keterangan
	Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran
	Pengambilan keputusan (<i>decision</i>)
	Layar peraga (<i>monitor</i>)
	Pemasukkan data secara manual

(Sumber: Kristanto, 2008:68-70)



2.2.5. Tahap Pengembangan Sistem

Kristanto (2011:37), tahap-tahap pengembangan sistem adalah sebagai berikut :

1. Penyelidikan Awal

Pada tahap penyelidikan awal, analisis belajar dari pemakai mengenai apa yang diharapkan dari sebuah sistem informasi yang baru. Hal-hal yang perlu diperhatikan dalam tahapan ini:

- a. Mencoba memahami dan menjelaskan apa yang diharapkan oleh pemakai (sistem informasi bagaimana yang mereka perlukan).
- b. Menentukan ruang lingkup dari studi sistem informasi.
- c. Menentukan kelayakan dari masing-masing alternatif dengan memperkirakan keuntungan kerugian yang didapat.

2. Studi Kelayakan

Studi kelayakan merupakan tahap yang paling penting, karena didalamnya menyangkut berbagai aspek sistem baru yang diusulkan. Laporan mengenai studi kelayakan harus disampaikan kepada manajemen, yang pada gilirannya akan memberikan beberapa perubahan, menyarankan untuk diadakan penelitian lebih mendalam atau memutuskan untuk segera dilaksanakan. Adapun tujuan dilaksanakannya studi kelayakan adalah sebagai berikut:

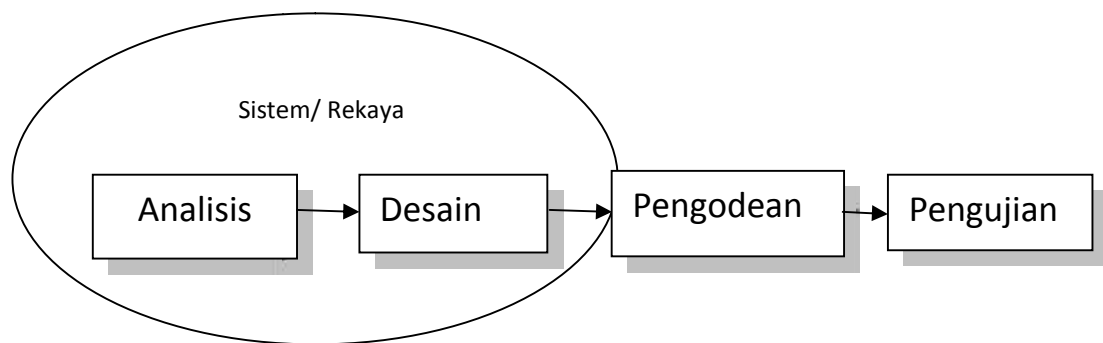
- a. Memperhitungkan sifat penyusunan sistem dengan memperhitungkan keberadaan masalah dan sifat masalah.
- b. Memperhitungkan jangkauan masalah.
- c. Mengajukan aksi-aksi yang dapat menyelesaikan masalah.
- d. Memperhitungkan kelayakan penyusunan sistem yang diajukan.
- e. Menyusun rencana detail untuk langkah analisis sistem.
- f. Menyusun rencana ringkasan untuk seluruh proyek penyusunan sistem.

2.2.6. Model *Waterfall*

Sukanto dan shalahuddin (2013:28) menyatakan, “Model SLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur



hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis desain, pengodean, pengujian, dan tahap pendukung (*support*). Berikut ini adalah gambar model air terjun:



Gambar 2.1. Model *Waterfall*

1. Analisis kebutuhan perangkat lunak
Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.
2. Desain
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, srditektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.
3. Pembuatan kode program
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.



4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2.7. Pengertian Kamus Data (*Data Dictionary*)

Ladjamudin (2013:70) mengemukakan “Kamus data sering disebut juga dengan sistem *data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi”.

Sukanto dan Shalahuddin (2013:73) menyatakan “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

1. Nama, nama dari data
2. Digunakan pada, merupakan proses–proses yang terkait data
3. Deskripsi, merupakan deskripsi data
4. Informasi tambahan, seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.



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

Tabel 2.8. Simbol-simbol Kamus Data(*Data Dictionary*)

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik ... atau...
{ } ⁿ	n kali diulang/bernilai banyak
()	Data opsional
...	Batas komentar

(Sumber : Sukamto dan Shalahuddin, 2013:74)

2.3. Teori Judul

2.3.1. Pengertian Pengolahan Data

Ladjamudin (2013:9) mendefinisikan “Pengolahan data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

2.3.2. Pengertian Pelayanan

Kamus Besar Bahasa Indonesia (2012:797) “Pelayanan adalah perihal atau cara melayani; usaha melayani kebutuhan orang lain dengan memperoleh imbalan (uang) atau jasa”.

2.3.3. Pengertian Teknik

Kamus Besar Bahasa Indonesia (2012:1422) “Teknik adalah pengetahuan dan kepandaian membuat sesuatu yang berkenaan dengan hasil industri (bangunan,



mesin); cara (kepandaian dsb) membuat atau melakukan sesuatu yang berhubungan dengan seni; metode atau sistem mengerjakan sesuatu”.

2.3.4. Pengertian Perseroan Terbatas (PT)

Kamus Besar Bahasa Indonesia (2012:1288) menyatakan, “Perseroan Terbatas (PT) merupakan perserikatan dagang atau perusahaan yang mempunyai modal bersama yang dibagi atas saham–saham (tanggung jawab pemegang saham terbatas sampai nilai nominal saham yang dimiliki)”.

2.3.5. Prima Persada Nusantara

Prima Persada Nusantara adalah perusahaan swasta nasional yang bergerak dalam bidang usaha pembangunan, perdagangan, konstruksi dan pengadaan jasa.

2.3.6. Pengertian Aplikasi Pengolahan Data Pelayanan Teknik pada PT Prima Persada Nusantara Kantor Cabang Palembang

Aplikasi Pengolahan Data Pelayanan Teknik pada PT Prima Persada Nusantara Kantor Cabang Palembang adalah sebuah aplikasi yang dibangun untuk mengolah data pelaksanaan pekerjaan pelayanan teknik semua rayon area Palembang setiap bulannya.

2.4. Teori Program

2.4.1. Sekilas tentang *Microsoft Visual Studio 2012*

Sujatmiko (2012:307), *Visual Basic* adalah bahasa komputer yang digunakan untuk membuat program interaktif di dalam sebuah *web page*, yang dikembangkan oleh perusahaan Microsoft Corp.

Supardi (2015:1) menyatakan, *Microsoft Visual Studio 2012* merupakan lingkungan terpadu dalam pembuatan program *IDE (Integrated Development Program)* buatan *Microsoft*. Terdapat dalam paket program *Visual Studio 2012* yang terintegrasi dengan program *Visual Basic 2012*, *Visual C++ 2012*, *Visual F#*, dan *Visual C# 2012*.

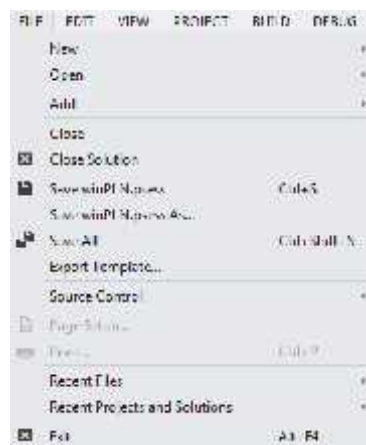


Pada *Visual Basic 2012* menggunakan *.Net Framework 4.5* dan sebagai *database default* (bawaan dari sistem) adalah *SQL Server 2012*. *Visual Basic 2012* merupakan *Visual Basic* pengembangan dari *Visual Basic 2010*.

2. 4.1.1. Antarmuka Menu *Microsoft Visual Studio 2012*

Menurut Winarno (2015: 37) ada beberapa menu di Visual Studio, diantaranya:

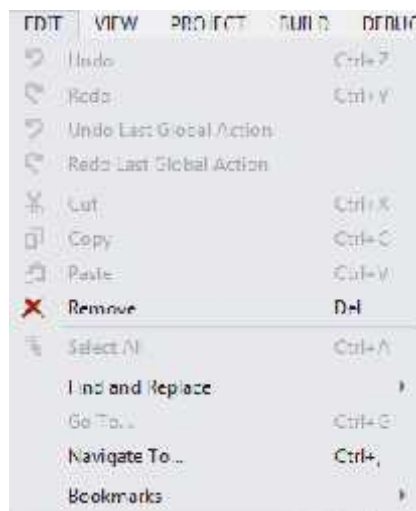
1. Menu File, menu ini berkaitan dengan pengolahan file. Beberapa menu penting adalah:
 - a. New Project: Membuat projek baru.
 - b. New Web site: Membuat website baru (misalnya menggunakan ASP.NET).
 - c. New Team Project: Membuka projek.
 - d. New File: Membuat file baru.
 - e. Open Project: Membuka projek.
 - f. Open Web site: Membuka website.
 - g. Open File: Membuka file baru.
 - h. Add: Menambahkan beberapa item tertentu.
 - i. Close: Menutup projek yang sedang terbuka.
 - j. Save: Menyimpan file.
 - k. Recent Files: Menampilkan MRU (most recently used) dan file.
 - l. Exit: Keluar dari Visual Studio.



Gambar 2.2. Menu File



2. Menu Edit, berisi beberapa menu yang berperan penting untuk editing file. Beberapa menu penting di menu ini adalah:
- Undo: Membatalkan action terakhir yang sudah dilakukan.
 - Redo: Mengembalikan action terakhir yang sudah dilakukan.
 - Cut: Memotong nilai untuk nanti bisa ditempelkan dengan Paste.
 - Copy: Menyalin nilai untuk bisa ditempelkan dengan Paste:
 - Paste: Menampilkan nilai dari clipboard yang diambil dari Cut atau Copy.
 - Delete: Menghapus nilai tertentu.
 - Select All: Memilih semua objek.
 - Find and Replace: Mencari teks dan menggantinya dengan teks yang lain.
 - Navigate To: Menavigasi ke bagian tertentu.
 - Bookmarks: Menandai bagian tertentu.



Gambar 2.3. Menu Edit

3. Menu View, berguna untuk mengakses dan melihat komponen-komponen tertentu. Beberapa menu lain dari menu ini adalah:
- Solution explorer: Membuka jendela Solution Explorer.
 - Team Explorer: Membuka jendela Team Explorer.
 - Server Explorer: Membuka jendela Server Explorer.
 - Bookmark Window: Membuka jendela Window.



- e. Call Hierarchy: Membuka jendela Call Hierarchy.
- f. Class View: Melihat jendela Class.
- g. Object Browser: Membuka jendela Object Browser.
- h. Error List: Menampilkan daftar error yang ada.
- i. Output: menampilkan jendela Output.
- j. Start Page: Membuka start page.
- k. Toolbox: Menampilkan toolbox.
- l. Full Screen: Menampilkan layar Visual Studio menjadi penuh.

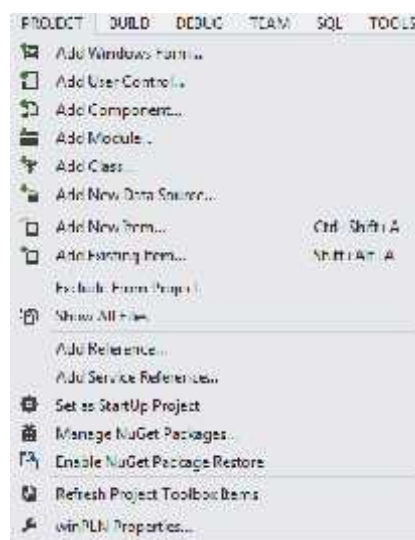


Gambar 2.4. Menu View

4. Menu Project, berkaitan dengan manajemen item-item di project. Beberapa menu penting di menu ini adalah:
 - a. Add Windows Form: Menambahkan form Windows baru ke dalam project.
 - b. Add User Control: Menambahkan control user baru ke dalam form.
 - c. Add Component: Menambahkan component baru ke dalam form.
 - d. Add Module: Menambahkan module baru.
 - e. Add Class: Menambahkan Class baru.

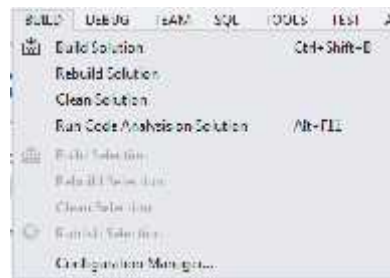


- f. Add New Data Source: Menambahkan data source baru.
- g. Add New Item: Menambahkan item baru ke project.
- h. Add Existing Item: Menambahkan item baru dari item yang sudah ada.
- i. Exclude From Project: Mengecualikan item dari project.
- j. Add Reference: Menambahkan referensi.
- k. Show All Files: Menampilkan semua file yang terlibat dalam proyek.
- l. Set as Startup Project: Menambahkan project startup
- m. Properties: Menampilkan properties dari Project.



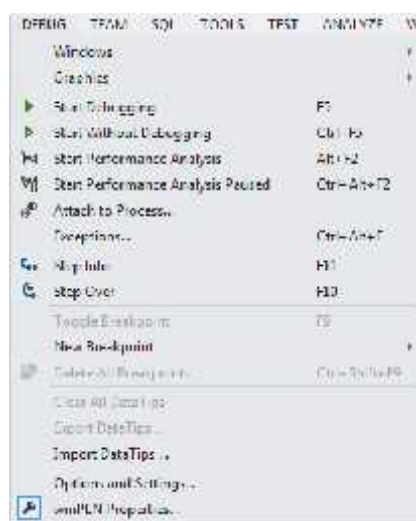
Gambar 2.5. Menu Project

5. Menu Build, digunakan untuk membangun solusi menjadi aplikasi. Berikut ini beberapa menu penting di menu ini:
 - a. Build Solution: Membangun file-file solusi project menjadi aplikasi.
 - b. Rebuild Solution: Membangun ulang solusi project menjadi aplikasi.
 - c. Clean Solution: Meng-clean solusi yang ada.
 - d. Build Project: Membangun project yang ada.
 - e. Configuration Manager: Membuka halaman manajer konfigurasi dari solusi.



Gambar 2.6. Menu Build

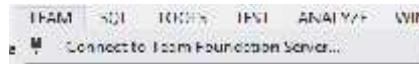
6. Menu Debug, untuk menjalankan program dan melakukan debugging untuk mencari error di program. Beberapa menu penting di sini adalah:
 - a. Start Debugging: Memulai proses debugging sambil menjalankan program.
 - b. Start Without Debugging: Menjalankan program tanpa melakukan debugging.
 - c. Exceptions: Menjalankan eksepsi.
 - d. Performance and Diagnostics: Mengecek performa dan mendiagnosis apabila ada masalah.
 - e. Step Into: Menuju ke bagian tertentu.
 - f. Step Over: Meloncati bagian tertentu.
 - g. New Breakpoint: Menambahkan breakpoint.



Gambar 2.7. Menu Debug



7. Menu Team, digunakan untuk mengakses fasilitas Team Foundation Server untuk membuat program kerjasama via tim.



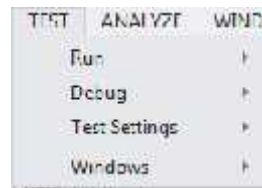
Gambar 2.8. Menu Team

8. Menu Tools, berisi tool-tool yang dapat dipakai untuk memperarah pembuatan program. Beberapa menu penting di menu ini adalah:
- Attach To Process: Melakukan attachment pada sebuah proses.
 - Connect to Database: Menghubungkan ke database untuk sumber data.
 - Connect to Server: Menghubungkan ke server.
 - Add in Manager: Manajer add-in.
 - Estensions and Updates: Melihat daftar ekstensi dan update yang ada di Visual Studio.
 - Options: Mengatur opsi dari berbagai aspek Visual Studio.



Gambar 2.9. Menu Tools

9. Menu Test, digunakan untuk melakukan pengetesan terhadap program. Beberapa menu penting disini adalah:
- Run: Menjalankan program.
 - Debug: Mendebug program.
 - Test Settings: Mengakses berbagai pengaturan tes.

**Gambar 2.10.** Menu Test

10. Menu Analyze: digunakan untuk menganalisis berbagai hal di Visual Studio.

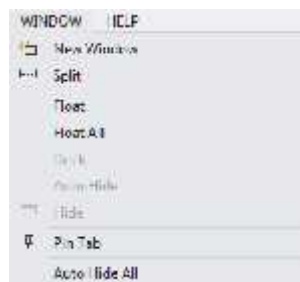
Beberapa menu di antaranya adalah:

- a. Performance and Diagnostics: Mengakses fasilitas pengecekan performa dan mendiagnosis kalau ada masalah.
- b. Profiler: Mengecek profil.
- c. Run Code: Menjalankan kode.

**Gambar 2.11.** Menu Analyze

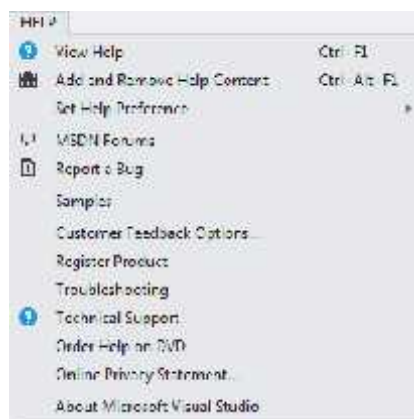
11. Menu Window: digunakan untuk mengakses dan mengatur berbagai jendela yang ada di Visual Studio. Beberapa menu pentingnya antara lain:

- a. New Window: Membuka window baru.
- b. Split window: Membagi window menjadi dua.
- c. Float: Membuat window jadi mengambang.
- d. Dock: membuat window menjadi ter-docking ke bagian tertentu.
- e. Auto Hide: Membuat window menjadi terlihat tapi tersembunyi jika tidak diakses.
- f. Auto Hide All: Membuat semuanya menjadi auto hide.



Gambar 2.12. Menu Window

12. Menu Help, digunakan untuk menampilkan berbagai bantuan.



Gambar 2.13. Menu Help

2.4.1.2. Panel-panel Jendela *Microsoft Visual Studio 2012*

Berikut ini beberapa panel jendela standar, diantaranya:

1. Data sources: Digunakan untuk menampilkan sumber database yang digunakan.
2. Toolbox: Berisi tool-tool untuk mengakses berbagai komponen atau objek yang bisa ditambahkan ke dalam form atau program.
3. Area Kerja: Bagian tengah yang menampilkan form yang sedang dibuat.
4. Output: Berisi pesan yang dikeluarkan oleh program.
5. Properties: Menampilkan beberapa properti dari objek tertentu.
6. Solution Explorer: Berisi komponen-komponenn yang ada di solusi yang sedang dibuka.
7. Team Explorer: Berisi user yang terhubung dalam tim.
8. Server Explorer: Berisi objek-objek yang ada di server.



2.4.2. Sekilas tentang *MySQL*

2.4.2.1. Pengertian *MySQL*

Achmad (2010:2) menyatakan, *MySQL* merupakan *software database open source* yang paling populer di dunia, dimana saat ini digunakan lebih dari 100 juta pengguna di seluruh dunia. Dengan kehandalan, kecepatan dan kemudahan penggunaannya, *MySQL* menjadi pilihan utama bagi banyak pengembang *software* dan aplikasi baik di *platform web* maupun *desktop*. Pengguna *MySQL* tidak hanya sebatas pengguna perseorangan maupun perusahaan kecil, namun perusahaan seperti *Yahoo!*, *Alcatel-Lucent*, *Google*, *Nokia*, *Youtube*, *Wordpress* dan *Facebook* juga merupakan pengguna *MySQL*".

Sidik (2012:313) menyatakan, "SQL adalah bahasa standar yang digunakan untuk melakukan akses dan manipulasi *database*, dalam hal ini *database relasional*".

2.4.2.2. Perintah-perintah SQL Dasar

1. *Select*

Select digunakan untuk membaca data dari suatu tabel atau *view*.

Sintaks:

Select namafield1, namafield2 ... from namatabel1 where kriteria

Atau

Select * from namatabel1 where kriteria

2. *Insert*

Digunakan untuk menambahkan data ke dalam tabel.

Sintaks:

Insert into namatabel (field1, ...) values ('nilai1', ...)

3. *Update*

Digunakan untuk memperbaiki data suatu field dalam suatu tabel.

Sintaks:

Update namatabel set field1=nilai, field2=nilai2... where kriteria

4. *Delete*

Digunakan untuk menghapus data dari suatu tabel.



Sintaks:

Delete from namatabel where kriteria

2.4.2.3. Fitur-fitur MySQL

Sebagai *software* DBMS, MySQL memiliki sejumlah fitur seperti dibawah ini:

1. Multiplatform

MySQL tersedia pada beberapa platform (Windows, Linux, Unix, dan lain-lain).

2. Andal, cepat, dan mudah digunakan

MySQL tergolong sebagai *database server* (server yang melayani permintaan terhadap database) yang andal, dapat menangani database yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses database, dan sekaligus mudah untuk digunakan.

3. Jaminan keamanan akses

MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan.

4. Dukungan SQL

MySQL mendukung perintal SQL (*Structured Query Language*). Sebagaimana diketahui, SQL merupakan standar dalam pengaksesan database relasional.

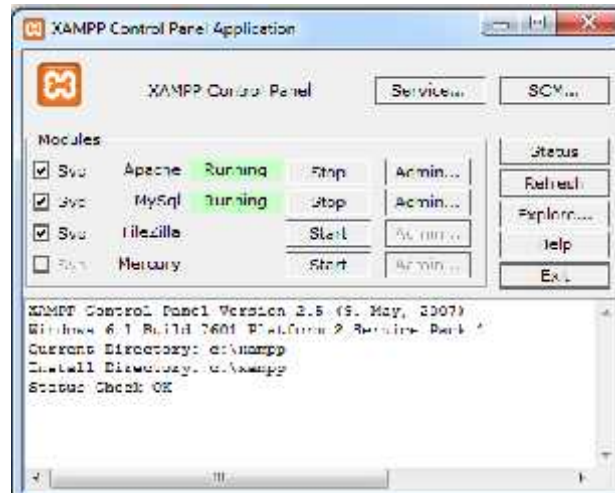
2.4.3. Sekilas tentang XAMPP

Sidik (2012:72) menyatakan, “XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) merupakan paket *server web* PHP dan databae MySQL yang paling populer di kalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai databasenya”.

Sebelum menjalankan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) yang harus dilakukan adalah memastikan bahwa Apache dan MySQL terkoneksi, dengan cara mengubah tombol *stop* pada action menjadi *start*. Berikut



adalah tampilan XAMPP (X(Windows/Linux) Apache MySQL PHP dan Perl) control panel untuk menjalankan Apache dan MySQL.



Gambar 2.14. Tampilan XAMPP Control Panel Application

Berikut ini adalah tampilan awal dari XAMPP:



Gambar 2.15. Tampilan Awal XAMPP