



---

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Teori Umum

##### 2.1.1. Pengertian Komputer

Febri (2012:3) Mengemukakan, "Komputer adalah peralatan yang menerima data dan menyimpan kemudian diproses untuk menghasilkan data dalam bentuk lain".

Com (2011:10) mengemukakan, "Komputer adalah serangkaian atau sekelompok mesin elektronik yang terdiri dari ribuan bahkan jutaan komponen yang saling bekerja sama, serta membentuk sebuah sistem yang rapi dan teliti".

Jadi, komputer adalah alat pemrosesan data elektronik yang membantu melakukan perintah-perintah atau proses-proses tertentu yang telah diurutkan atau diprogramkan.

##### 2.1.2. Pengertian Perangkat Lunak

Sukanto dan Shalahuddin (2014:2), "Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)".

Sujatmiko (2013:201), "Perangkat lunak adalah istilah umum untuk data yang diinformasi dan disimpan secara digital, termasuk program komputer, dokumentasinya, dan berbagai informasi yang bisa dibaca dan ditulis oleh komputer".

Jadi, perangkat lunak atau *software* merupakan program komputer terasosiasi dengan dokumen perangkat lunak yang memiliki serangkaian aturan tertentu dalam pengoperasiannya.

##### 2.1.3. Pengertian Data

Kristanto (2011:7), "Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata".



---

Al-bahra (2013:20), "Data adalah komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi".

Febri (2012:3), "Data adalah suatu file atau pun field yang berupa karakter atau tulisan dan gambar".

Jadi, data adalah kumpulan dari angka maupun karakter yang berasal dari kenyataan yang belum berarti dan harus diolah terlebih dahulu untuk menjadi suatu informasi yang lebih berarti bagi penggunaannya.

#### **2.1.4. Pengertian Sistem**

Hartono (2013:9), "Sistem adalah suatu himpunan dari berbagai bagian atau elemen yang saling berhubungan secara terorganisasi berdasar fungsi-fungsinya menjadi satu kesatuan".

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.

##### **2.1.4.1. Metode Pengembangan Sistem**

Sukanto dan Shalahudin (2014:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode 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 terurut mulai dari analisis, desain, pengodean, pengujian, dan pemeliharaan.

###### **a. Analisis**

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

###### **b. Desain**

Tahap desain adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi

---



antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

#### c. Pengkodean

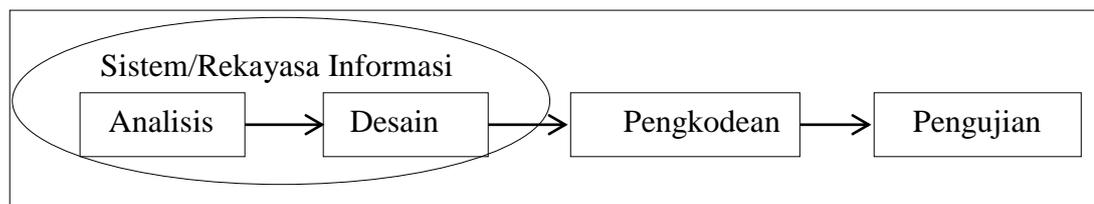
Pada tahap pengkodean, desain harus ditranslasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

#### d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika 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.

#### e. Pemeliharaan

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



**Gambar 2.1.** Ilustrasi Model *Waterfall*

(Sumber : Sukamto dan Shalahudin, 2014 : 28)



## **2.2. Teori Khusus**

### **2.2.1. Pengertian Basis Data (*Database*)**

Kadir (2013: 15), “Database adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data”.

Yakub (2012:51), “Basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi). Relasi biasanya ditunjukkan dengan kunci (*key*) dari tiap file yang ada”.

Sukanto dan Shalahuddin (2013:43) menyatakan “Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi tersedia saat dibuat”.

Sujatmiko (2013:76) menyatakan “Basis data merupakan representasi kumpulan data yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan”.

Jadi, basis data (*database*) adalah kumpulan data yang saling berhubungan, tersimpan di komputer, kemudian diolah menjadi suatu informasi yang berguna.

### **2.2.2. Pengertian Kamus Data**

Sukanto dan Shalahuddin (2013:73), “Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap file atau file di dalam sistem”.

Al-Bahra (2013 : 70 ),”Kamus data adalah katalog fakta tentang data dan kebutuhan”.

Kristanto (2013:50), menjelaskan simbol-simbol yang di gunakan dalam kamus data, yaitu :

**Tabel 2.2.** Simbol-simbol dalam Kamus Data

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

(Sumber : Sukanto dan Shalahuddin 2013:50)

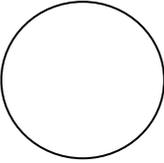
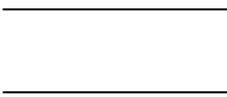
### 2.2.3. Pengertian Data Flow Diagram (DFD)

Kristanto, (2011:55), “*Data Flow Diagram (DFD)* adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.

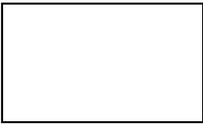
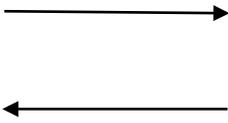
Al-Bahra (2013:64) mendefinisikan bahwa, “*DFD* adalah model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil”.

Sukanto dan Shalahuddin, (2014:71), menjelaskan notasi pada *DFD* adalah sebagai berikut :

Tabel 2.3. Simbol-simbol *Data Flow Diagram*

NO	SIMBOL	KETERANGAN
1		<p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan Diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>
2		<p><i>File</i> 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 (ERD), Conceptual Data Model (CDM), Physical Data Model (PDM)</i>)</p> <p>Catatan :</p> <p>Nama yang diberikan pada sebuah penyimpanan biasanya kata benda</p>

Lanjutan Tabel 2.3. Simbol-simbol *Data Flow Diagram*

NO	SIMBOL	KETERANGAN
3.		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> ) berupa kata benda.
4.		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”

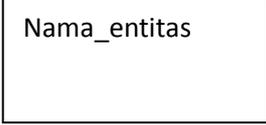
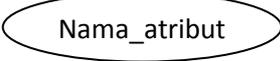
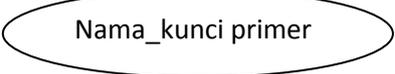
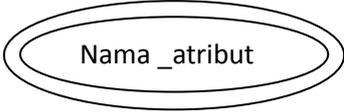
(Sumber : Sukamto dan Shalahuddin, 2014:71)

#### 2.2.4. Pengertian ERD (*Entity Relational Diagram*)

Yakub (2012:60), “*Entity Relationship Diagram (ERD)* merupakan suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara abstrak”.

Adapun simbol-simbol yang digunakan dalam ERD, yaitu:

**Tabel 2.4.** Simbol-simbol *Entity Relationship Diagram (ERD)*

NAMA	SIMBOL	KETERANGAN
Entitas/ <i>entity</i>		Merupakan data inti yang akan disimpan sehingga dapat diakses oleh aplikasi komputer.
Atribut		Merupakan <i>field</i> atau kolom data yang akan disimpan dalam suatu entitas.
Atribut kunci primer		Merupakan <i>field</i> atau kolom data yang disimpan dalam suatu entitas serta menggunakan kunci akses <i>record</i> yang diinginkan ataupun unik.
Atribut multinilai/ <i>multivalued</i>		Merupakan <i>field</i> atau kolom data yang disimpan dalam suatu entitas dimana memiliki nilai lebih dari satu.
Relasi		Merupakan relasi yang menghubungkan antar entitas dan biasanya diawali dengan kata kerja.



**Lanjutan Tabel 2.4.** Simbol-simbol *Entity Relationship Diagram (ERD)*

NAMA	SIMBOL	KETERANGAN
Asosiasi/ <i>association</i>		Merupakan penghubung antara relasi dimana kedua ujungnya memiliki <i>multiplicity</i> dalam jumlah pemakaian.

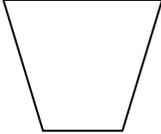
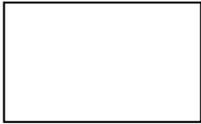
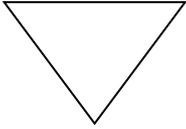
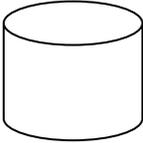
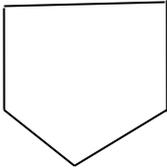
(Sumber: Sukanto dan Shalahuddin 2013:50)

### 2.2.5. Pengertian *Blokchart*

Kristanto (2011:68) menjelaskan, “*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”.

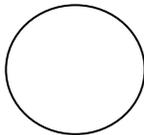
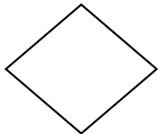
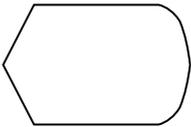
Kristanto (2011:68) menjelaskan, “Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

**Tabel 2.4.** Simbol-simbol dalam *Block Chart*

NO	SIMBOL	KETERANGAN
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen
3.		Proses Manual
4.		Proses dilakukan oleh komputer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan ( <i>Storage</i> )
7.		Proses apa saja yang tidak <i>terDefinisi</i> termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.



**Lanjutan Tabel 2.4.** Simbol-simbol dalam *Block Chart*

NO	SIMBOL	KETERANGAN
9.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal yang menandakan awal dan akhir dari suatu aliran.
11.		Pengambilan keputusan ( <i>Decision</i> ).
12.		Layar peraga ( <i>monitor</i> ).
13		Pemasukkan data secara manual.

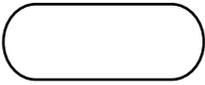
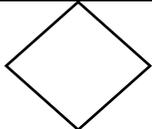
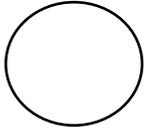
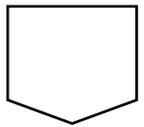
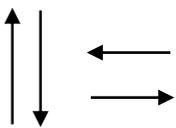
(Sumber : Kristanto, 2011 : 68)

### 2.2.6. Pengertian *Flowchart*

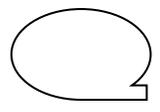
*Ewolf Community* (2012:16) mengemukakan, “*Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya”.

*Ewolf Community* (2012:16) juga menjelaskan tentang simbol-simbol yang sering digunakan dalam *flowchart* dapat dilihat pada tabel berikut ini:

**Tabel 2.5.** Simbol-simbol dalam *Flow Chart*

NO	SIMBOL	ARTI
1.		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu.
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar simbol
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll

**Lanjutan Tabel 2.5.** Simbol-simbol dalam *Flow Chart*

NO	SIMBOL	KETERANGAN
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau ke sebuah pita <i>magnetic</i>
13.		Simbol <i>database</i> atau basis data

(Sumber :Ewolf community,2012:16)

### 2.2.7. Pengertian Daftar Kejadian (*Event List*)

Kristanto (2011:64) menjelaskan bahwa daftar kejadian (*event list*) digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Adapun cara mendeskripsikan daftar kejadian adalah sebagai berikut :

1. Pelaku adalah *entity* luar, jadi bukan sistem.



2. Menguji setiap *entity* luar dan mencoba mengevaluasi setiap *entity* luar yang terjadi pada sistem.
3. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
4. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan *entity* luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respon dari kejadian.
6. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
7. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

### 2.3. Teori Judul

#### 2.3.1. Pengertian Aplikasi

Kamus Besar Bahasa Indonesia (81), “Aplikasi adalah program komputer atau perangkat lunak yang didesain untuk mengerjakan tugas tertentu”.

Sujatmiko (2013:23) berpendapat “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas tertentu, misalnya Microsoft Word dan Microsoft Excel”.

Sutabri (2012:146), berpendapat “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimiliki”.

Dari kedua pengertian di atas, dapat disimpulkan bahwa Aplikasi adalah program komputer atau *software* yang dibuat oleh suatu perusahaan komputer untuk membantu manusia untuk mengerjakan tugas-tugas tertentu.

#### 2.3.2. Pengertian Laporan

Kamus Besar Bahasa Indonesia (365), “Laporan adalah segala sesuatu yang disampaikan, yang diadakan dan dilaporkan”.



### **2.3.3. Pengertian Rencana**

Kamus Besar Bahasa Indonesia (515), “Rencana adalah rancangan, rangka sesuatu yang harus dikerjakan, laporan pemberitaan, acara, program dan artikel”.

### **2.3.4. Pengertian Anggaran**

Kamus Besar Bahasa Indonesia (39), “Anggaran adalah rencana penjatahan sumber daya yang dinyatakan dengan angka, biasanya dalam satuan uang, perkiraan, perhitungan, aturan, taksir mengenai penerimaan dan pengeluaran kas yang diharapkan untuk periode mendatang”.

### **2.3.5. Pengertian Biaya**

Kamus Besar Bahasa Indonesia (102), “Biaya adalah dana yang dipakai untuk keperluan sesuatu”.

### **2.3.6. Pengertian Penawaran**

Kamus Besar Bahasa Indonesia (576), “Penawaran adalah saling meminta pengurangan harga”.

### **2.3.7. Pengertian Tender**

Kamus Besar Bahasa Indonesia(580), “Tender adalah tawaran untuk mengajukan harga, memborong pekerjaan, atau menyediakan barang hanya pengusaha bermodal besar yang mampu mengikutinya”.

### **2.3.8. Pengertian Proyek**

Kamus Besar Bahasa Indonesia(494), “Proyek adalah rencana pekerjaan dengan sasaran khusus (pengairan, pembangkit listrik, dan sebagainya) dan dengan saat penyelesaian dengan tegas”.

Sutabri (2012:151),”Proyek adalah rencana pekerjaan dengan sasaran khusus dan dengan saat pelaksanaan pekerjaan yang tegas”.

Dari kedua pengertian di atas, dapat disimpulkan bahwa Proyek adalah rencana pekerjaan dan pelaksanaan pekerjaan yang tegas.

---



### 2.3.9. Pengertian Konstruksi

Kamus Besar Bahasa Indonesia (340), “Konstruksi adalah disebut kontraktor yaitu pemborong”.

### 2.3.10. Pengertian Aplikasi Laporan Rencana Anggaran Biaya (RAB)

#### Penawaran Tender Proyek Konstruksi pada CV Anco Jaya Palembang

Aplikasi Laporan Rencana Anggaran Biaya (RAB) Penawaran Tender Proyek Konstruksi pada CV Anco Jaya Palembang adalah suatu aplikasi yang dibangun untuk proses rencana anggaran biaya proyek pembangunan yang dibuat kontraktor untuk memperkirakan berapa sebenarnya biaya sesungguhnya yang dibutuhkan untuk menyelesaikan suatu kontrak kerja proyek konstruksi yang bertujuan untuk dapat menentukan budget dana yang harus disediakan dengan menggunakan aplikasi *Visual Basic.NET* dengan menggunakan Database *MySQL XAMPP*.

## 2.4. Teori Program

### 2.4.1. Pengertian *MySQL*

Kadir (2008:348), “*MySQL* adalah salah satu jenis database server yang sangat terkenal. Kepopulerannya disebabkan *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses databasenya. Selain itu, ia bersifat open source pada berbagai platform”.

Kadir (2013 :15) ,”*MySQL* adalah nama databases server. Database server adalah server yang berfungsi untuk menanggapi database.dengan menggunakan *MySQL* kita bisa menyimpan data dan kemudian data bisa di akses dengan cara mudah dan cepat”.

Dari kedua pengertian di atas, dapat disimpulkan bahwa *MySQL* adalah jenis *software* pada *Relational Database Management System(RDBMS)* pada sebuah manajemen *database* sebagai basis data.



## **2.4.1. Sekilas tentang Microsoft Visual Basic Studio 2012**

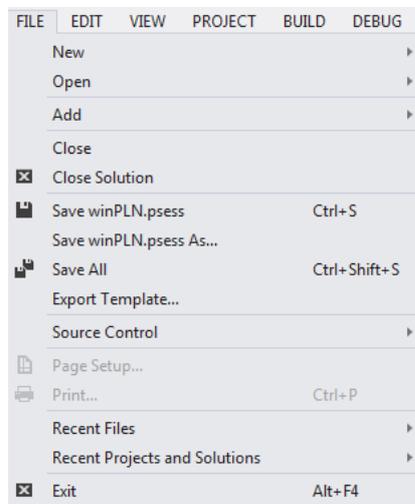
### **2.4.2.1. Pengertian Microsoft Visual Studio 2012**

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*.

### **2.4.2.2. Antarmuka Menu Microsoft Visual Studio 2012**

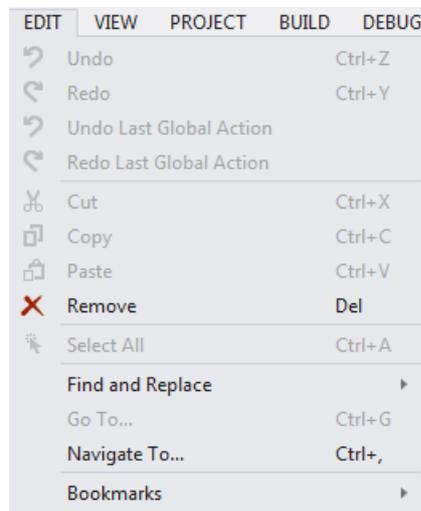
Winarno (2015 : 35-51), Ada beberapa menu di Visual Studio, diantaranya:

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.



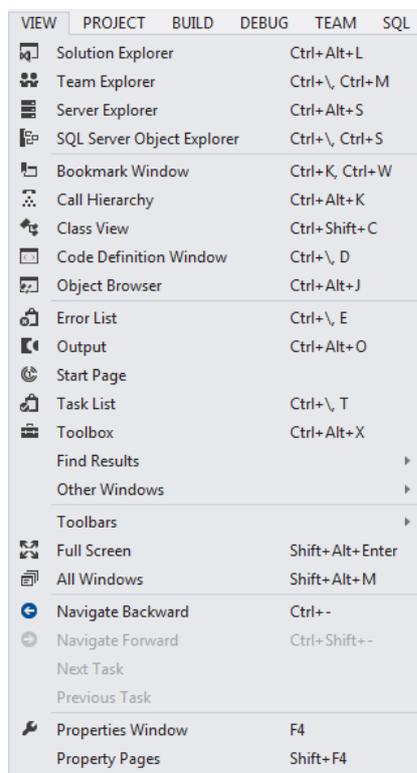
**Gambar 2.2.** Menu *File*

2. Menu Edit, berisi beberapa menu yang berperan penting untuk editing file. Beberapa menu penting di menu ini adalah:
  - a. Undo: Membatalkan action terakhir yang sudah dilakukan.
  - b. Redo: Mengembalikan action terakhir yang sudah dilakukan.
  - c. Cut: Memotong nilai untuk nanti bisa ditempelkan dengan Paste.
  - d. Copy: Menyalin nilai untuk bisa ditempelkan dengan Paste:
  - e. Paste: Menampilkan nilai dari clipboard yang diambil dari Cut atau Copy.
  - f. Delete: Menghapus nilai tertentu.
  - g. Select All: Memilih semua objek.
  - h. Find and Replace: Mencari teks dan menggantinya dengan teks yang lain.
  - i. Navigate To: Menavigasi ke bagian tertentu.
  - j. 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:
  - a. Solution explorer: Membuka jendela Solution Explorer.
  - b. Team Explorer: Membuka jendela Team Explorer.
  - c. Server Explorer: Membuka jendela Server Explorer.
  - d. 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.

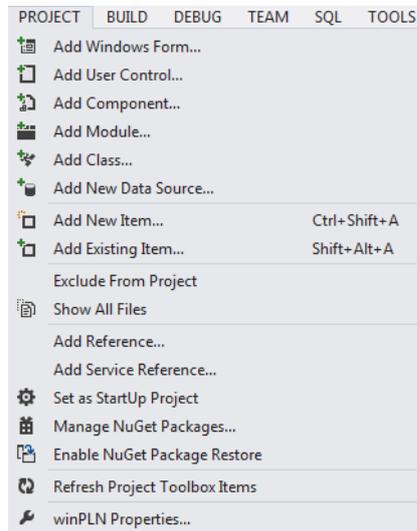


**Gambar 2.4.** Menu *View*

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 projek.

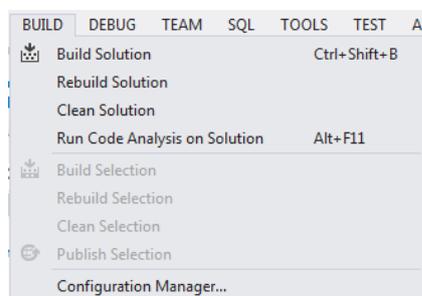


- l. Set as Startup Project: Menambahkan project startup
- m. Properties: Menampilkan properties dari Project.



**Gambar 2.5.** Menu Project

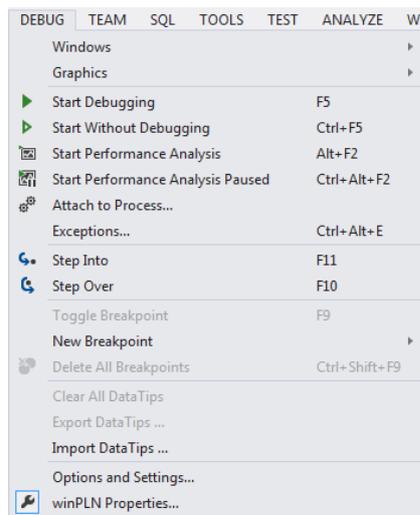
5. Menu Build, digunakan untuk membangun solusi menjadi aplikasi. Berikut ini beberapa menu penting di menu ini:
  - a. Build Solution: Membangun file-file solusi project menjadi aplikasi.
  - b. Rebuild Solution: Membangun ulang solusi project menjadi aplikasi.
  - c. Clean Solution: Meng-clean solusi 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*

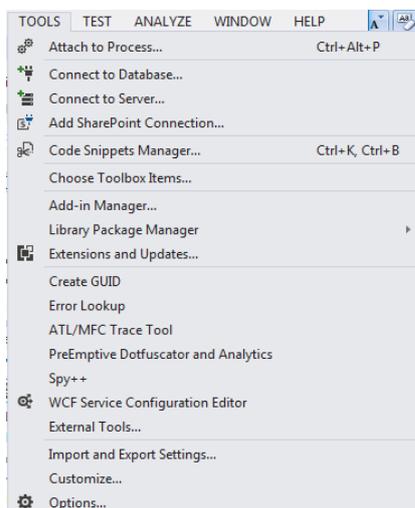
7. Menu Team, digunakan untuk mengakses fasilitas Team Foundation Server untuk membuat program kerjasama via tim.



**Gambar 2.8.** Menu *Team*

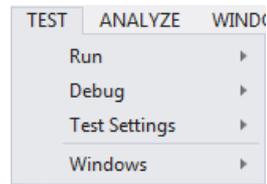


8. Menu Tools, berisi tool-tool yang dapat dipakai untuk memperarah pembuatan program. Beberapa menu penting di menu ini adalah:
  - a. Attach To Process: Melakukan attachment pada sebuah proses.
  - b. Connect to Database: Menghubungkan ke database untuk sumber data.
  - c. Connect to Server: Menghubungkan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



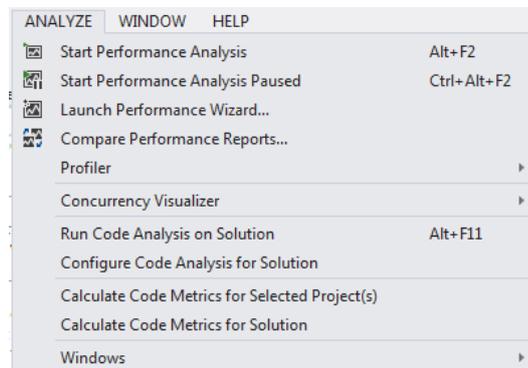
**Gambar 2.9.** Menu Tools

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



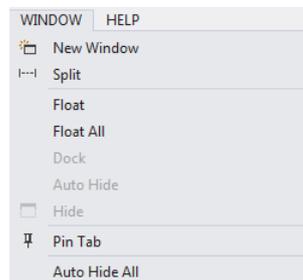
**Gambar 2.10.** Menu Test

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.



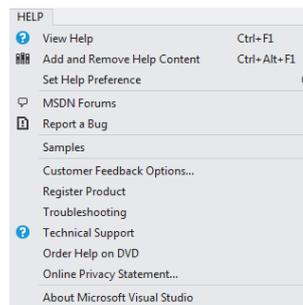
**Gambar 2.11.** Menu *Analyze*

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

#### 2.4.2.3. 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.
-



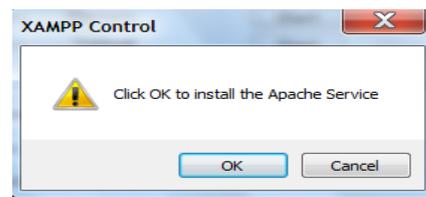
#### 2.4.2.4. Sekilas tentang XAMPP

Nugroho (2013:1), Mengemukakan ”XAMPP adalah paket program web lengkap yang dapat dipakai untuk belajar pemrogram web khususnya PHP dan MySQL.

Adapun langkah-langkah pemasangan XAMPP dari file zip yang telah di download adalah sebagai berikut :

1. Unzip file XAMPP yang telah didownload, ke dalam file directori c:/xampp
2. Masuk ke dalam directori c:/xampp
3. Pilih file xampp-control.exe, kemudian jalankan dengan menekan <enter> atau klik dua kali, tampilan berikut akan ditampilkan :

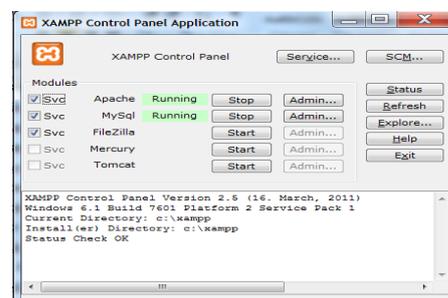
pada baris *Svc Apache*, dialog konfirmasi unjuk menjalankan Apache menjadi *Service* ditampilkan seperti berikut :



**Gambar 2.14.** Tampilan Dialog Konfirmasi Menjalankan *Apache*

(*Sumber : Nugroho 2013:6*)

4. Klik tombol OK.
5. Setelah itu klik tombol start pada baris yang sama, menjalankan Apache. Maka tampilan menjadi seperti berikut :



**Gambar 2.15.** Tampilan *Svc Apache Running*

(*Sumber : Nugroho (2013:6)*)