

## BAB III

### LANDASAN TEORI

#### 3.1 Visual Basic 6.0

Visual Basic merupakan cara termudah dan tercepat untuk membuat aplikasi yang dijalankan di sistem operasi Microsoft Windows®. Apakah Anda seorang profesional atau pemula sekalipun di bidang pemrograman Windows, kata “Visual” merujuk kepada metode yang digunakan untuk membuat antar muka yang bersifat grafis *Graphical User Interface* (GUI). Daripada menulis berbaris-baris kode untuk menjelaskan pemunculan dan lokasi dari suatu elemen di dalam antar muka, Anda dengan mudah dapat menambahkan object yang sebelumnya sudah dibangun ke dalam tempat dan posisi yang Anda inginkan di layar Anda. Jika Anda pernah menggunakan program untuk menggambar seperti *Paint*, maka Anda sebenarnya sudah memiliki keahlian untuk membuat sebuah antar muka pengguna secara efektif. Kata “Basic” merujuk kepada bahasa BASIC (*Beginners All-Purpose Symbolic Instruction Code*), sebuah bahasa yang digunakan oleh banyak programmer dibandingkan dengan bahasa lainnya dalam sejarah komputer. Visual *Basic* telah berubah dari bahasa asli BASIC dan sekarang memiliki ratusan pernyataan (*statements*), fungsi (*functions*), dan kata kunci (*keywords*), dan kebanyakan di antaranya terkait dengan antar muka grafis di Windows. Pengguna tingkat pemula sekalipun dapat membuat aplikasi dengan mempelajari hanya beberapa kata kunci, sementara kekuatan dari bahasanya membolehkan para pengguna tingkat profesional mencapai apapun yang dapat dihasilkan dengan menggunakan bahasa pemrograman Windows lainnya. Bahasa pemrograman Visual Basic tidaklah hanya identik dengan Visual Basic saja.

Sistem Pemrograman Visual Basic dalam bentuk Edisi Aplikasi, telah dimasukkan ke dalam Microsoft Excel, Microsoft Access, dan banyak aplikasi Windows lainnya juga menggunakan bahasa yang sama. *Visual Basic Scripting Edition* (VBScript) adalah sebuah bahasa skrip yang digunakan secara lebih umum dan merupakan bagian dari bahasa Visual Basic. Dengan mempelajari Visual Basic, maka Anda akan dibawa ke area-area yang telah disebutkan tadi. Dalam pengaplikasian program Visual Basic ada lima pokok yang penting di dalam komponennya. yaitu :

A. *Project* : *Project* adalah nama file yang akan dikenal dalam pemanggilan program aplikasi.

B. *Sub Form/MDI Form* : *Form* utama dalam Microsoft Visual Basic. karena *MDI form* ini mewakili *form - form* yang ada di dalam aplikasi *database* yang kita buat.

C. *Form* : untuk pengimplementasian sistem yang dapat melakukan beberapa perintah eksekusi sesuai yang diinginkan. dalam *form* ini juga kita dapat mengkodekan aplikasi untuk memanggil *database*, menampilkan data, menghapus data, memperbarui data, mengedit data dan mencetak data yang telah kita eksekusi.

D. *Coding/syntax* : perintah-perintah dalam bahasa program aplikasi Visual Basic yang telah ditentukan. Dalam *coding* ini juga yang mempengaruhi sistem dapat berjalan atau tidak.

E. *Report* : Hasil cetak yang diinginkan *user* yang digunakan sebagai *output* dari program.

F.Module : Suatu perintah untuk mengkoneksikan antara program Visual Basic dengan *database* yang akan di gunakan di dalam mengaplikasikan program.<sup>[3][7]</sup>

Variabel adalah lokasi untuk menyimpan data. Sebuah variabel hanya bisa menampung satu tipe data. Suatu program dapat memiliki banyak variabel sesuai yang dibutuhkan, tetapi sebelum dapat digunakan variabel tersebut harus dideklarasikan terlebih dahulu. Deklarasi diawali dengan DIM untuk mendeklarasikan variabel (di mana DIM singkatan dimensi). Berikut ini adalah

contoh format deklarasi variable :

```
Dim namavariabel As tipe data
```

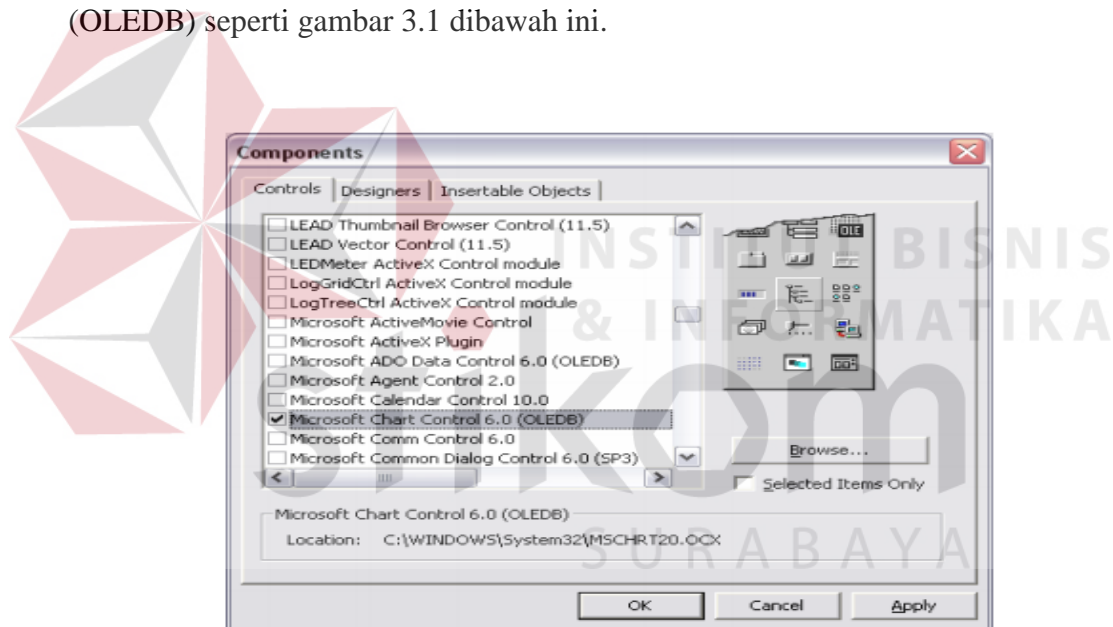


<b>Data type</b>	<b>Storage size</b>	<b>Range</b>
<b>Byte</b>	1 byte	0 to 255
<b>Boolean</b>	2 bytes	<b>True</b> or <b>False</b>
<b>Integer</b>	2 bytes	-32,768 to 32,767
<b>Long</b> (long integer)	4 bytes	-2,147,483,648 to 2,147,483,647
<b>Single</b> (single-precision floating-point)	4 bytes	-3.402823E38 to -1.401298E-45 for negative values; 1.401298E-45 to 3.402823E38 for positive values
<b>Double</b> (double-precision floating-point)	8 bytes	-1.79769313486231E308 to -4.94065645841247E-324 for negative values; 4.94065645841247E-324 to 1.79769313486232E308 for positive values
<b>Currency</b> (scaled integer)	8 bytes	-922,337,203,685,477.5808 to 922,337,203,685,477.5807
<b>Decimal</b>	14 bytes	+/-79,228,162,514,264,337,593,543,950,335 with no decimal point; +/-7.9228162514264337593543950335 with 28 places to the right of the decimal; smallest non-zero number is +/-0.00000000000000000000000000000001
<b>Date</b>	8 bytes	January 1, 100 to December 31, 9999
<b>Object</b>	4 bytes	Any <b>Object</b> reference
<b>String</b> (variable-length)	10 bytes + string length	0 to approximately 2 billion
<b>String</b> (fixed-length)	Length of string	1 to approximately 65,400
<b>Variant</b> (with numbers)	16 bytes	Any numeric value up to the range of a <b>Double</b>
<b>Variant</b> (with characters)	22 bytes + string length	Same range as for variable-length <b>String</b>

User-defined (using <b>Type</b> )	Number required by elements	The range of each element is the same as the range of its data type.
-----------------------------------	-----------------------------	--

### 3.2 MSChart

Adalah sebuah komponen yang terdapat pada Visual Basic 6.0 yang digunakan untuk menampilkan data dalam bentuk grafik. Pengambilan komponen MSChart dapat dilakukan dengan cara menambah komponen MSChart pada *Project*, menu *Project >> Components*. Pilih Microsoft Chart Control 6.0 (OLEDB) seperti gambar 3.1 dibawah ini.



**Gambar 3.1 Penambahan komponen MSChart pada VB 6.0**

Terdapat beberapa *property* yang harus diatur kedalam MSChar untuk membuat sebuah grafik. *User* hanya perlu mengaturnya dalam properti MSChart atau mengaturnya didalam program. Properti – properti yang harus diatur diantaranya adalah :

### 1. *Column Count*

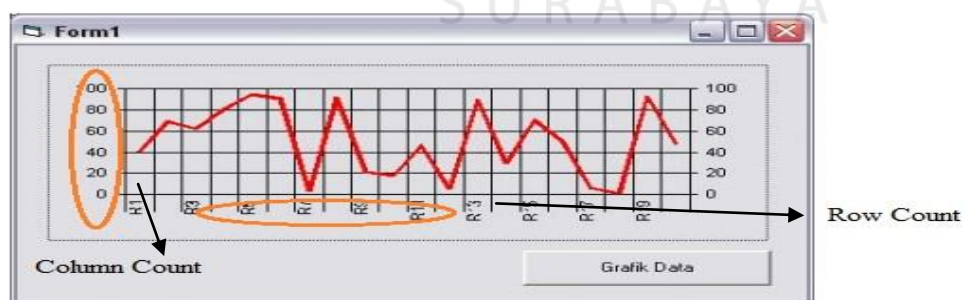
Digunakan sebagai acuan nilai dari data yang akan dimasukkan kedalam sebuah grafik. Misalnya saja tahun dan banyaknya penduduk sebuah provinsi. ( misalnya saja dalam grafik kepadatan penduduk pulau Jawa. Yang dijadikan sebagai *column count* adalah jumlah kepadatan penduduk ). Tapi hal ini juga tergantung dari kebutuhan *user* dalam mengolah data.

### 2. *Row Count*

Digunakan sebagai acuan penambahan data. ( misalnya saja dalam grafik kepadatan penduduk pulau Jawa. Yang dijadikan sebagai *row count* adalah tahun dari data). Tapi hal ini juga tergantung dari kebutuhan *user* dalam mengolah data.

### 3. *Chart Type*

Terdapat beberapa tipe grafik yang dapat dipilih di MSChart, misalnya saja grafik batang, grafik garis, grafik point, diagram lingkaran, panah, dan masih banyak lagi yang dapat dipilih oleh *user* sesuai dengan kebutuhan *user*. Bentuk dari chart juga dapat dalam bentuk 3D maupun 2D.



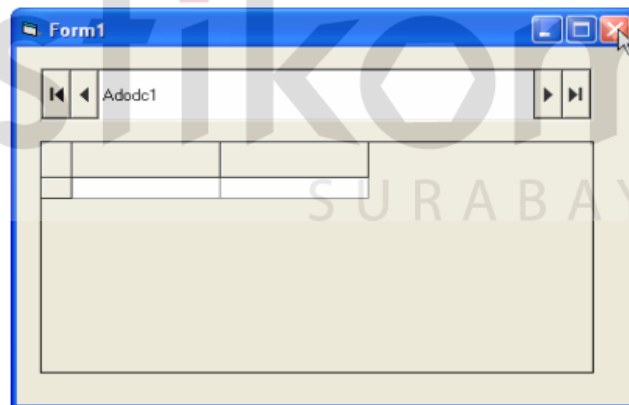
**Gambar 3.2 Hasil grafik pada MSChart**

### 3.3 Koneksi antara database dengan Visual Basic 6.0

Terdapat beberapa cara dalam membuat koneksi antara *database* terutama dalam hal ini adalah Microsoft Access dengan Visual Basic. Diantaranya adalah dengan menggunakan komponen “ADODC” yang terdapat pada Visual Basic, atau menggunakan koneksi “ODBC” tanpa melalui “DSN”, juga dapat melalui model “DSN” dengan menggunakan objek “ODBC”. *User* dapat menentukannya sesuai dengan kebutuhan dan juga efektifitas sebuah aplikasi. Karena setiap cara memiliki kekurangan dan kelebihan masing – masing.

#### 1. Menggunakan ADODC

Dengan cara menambahkan komponen ADODC yang terdapat pada Visual Basic 6.0. Biasanya *database* dapat terkoneksi langsung pada komponen DataGrid yang ada pada visual basic dengan cara mengaturnya pada properti DataGrid. Pengaturan dilakukan pada properti ADODC.



**Gambar 3.3 Contoh penggunaan ADODC**

#### 2. Menggunakan koneksi ADODB tanpa DSN

Penggunaan metode ADODB tanpa DSN dapat dilakukan dengan cara membuat program koneksi pada *database* pada modul Visual Basic agar dapat diakses oleh semua *form* pada Visual Basic. Dan koneksi dapat dilakukan pada sintaks program Visual Basic dibawah ini.

```

Sub koneksi(flag As Boolean)
Set conn = New ADODB.Connection

On Error GoTo Handler
If flag = True Then
    conn.Open "PROVIDER=MICROSOFT.JET.OLEDB.4.0;PERSIST
              SECURITY INFO=FALSE;DATA SOURCE=" & App.Path &
              "\batan.mdb"
    conn.CursorLocation = adUseClient
Else
    conn.Close
End If
Exit Sub

Handler:
    MsgBox Err.Description, vbCritical, "Error"
End Sub

```

### 3.4 Metode pengolahan data

Nilai IKM dihitung dengan menggunakan “nilai rata-rata tertimbang” masing-masing unsur pelayanan. Dalam penghitungan indeks kepuasan masyarakat terhadap 14 unsur pelayanan yang dikaji, setiap unsur pelayanan memiliki penimbang yang sama dengan rumus sebagai berikut:

$$\text{Bobot nilai rata-rata tertimbang} = \frac{\text{Jumlah bobot}}{\text{Jumlah Unsur}} = \frac{1}{14} = 0,071$$

Untuk memperoleh nilai IKM unit pelayanan digunakan pendekatan nilai rata-rata tertimbang dengan rumus sebagai berikut:

$$\text{IKM} = \frac{\text{Total dari Nilai Persepsi Per Unsur}}{\text{Total unsur yang terisi}} \times \text{Nilai penimbang}$$

Untuk memudahkan interpretasi terhadap penilaian IKM yaitu antara 25 – 100 maka hasil penilaian tersebut diatas dikonversikan dengan nilai dasar 25, dengan rumus sebagai berikut:

$$\text{IKM Unit Pelayanan} \times 25$$



Mengingat unit pelayanan mempunyai karakteristik yang berbeda-beda, maka setiap unit pelayanan dimungkinkan untuk:

- a. Menambah unsur yang dianggap relevan.
- b. Memberikan bobot yang berbeda terhadap 14 (empat belas) unsur yang dominan dalam unit pelayanan, dengan catatan jumlah bobot seluruh unsur tetap 1.

Tabel: Nilai Persepsi, Interval IKM, Interval Konversi IKM, Mutu Pelayanan dan Kinerja Unit Pelayanan

NILAI PERSEPSI	NILAI INTERVAL IKM	NILAI INTERVAL KONVERSI IKM	MUTU PELAYANAN	KINERJA UNIT PELAYANAN
1	1,00 – 1,75	25 – 43,75	D	Tidak baik
2	1,76 – 2,50	43,76 – 62,50	C	Kurang baik
3	2,51 – 3,25	62,51 – 81,25	B	Baik
4	3,26 – 4,00	81,26 – 100,00	A	Sangat baik

### 3.5 Perangkat pengolahan

#### 3.5.1 Pengolahan dengan computer

Data entry dan penghitungan indeks dapat dilakukan dengan program komputer/ sistem data base.

#### 3.5.2 Pengolahan secara manual

1. Data Isian kuesioner dari setiap responden dimasukkan ke dalam formulir mulai dari unsur 1 ( U1) sampai dengan unsur 14 (U14);
2. Langkah selanjutnya untuk mendapatkan nilai rata-rata per unsur pelayanan dan nilai indeks unit pelayanan adalah sebagai berikut:
  - a) Nilai rata-rata per unsur pelayanan.

Nilai masing-masing unsur pelayanan dijumlahkan (kebawah) sesuai dengan jumlah kuesioner yang diisi oleh responden, kemudian untuk mendapatkan nilai rata-rata per unsur pelayanan, jumlah nilai masing-masing unsur pelayanan dibagi dengan jumlah responden yang mengisi. Untuk mendapatkan nilai rata-rata tertimbang per unsur pelayanan, jumlah nilai rata-rata per unsur pelayanan dikalikan dengan 0,071 sebagai nilai bobot rata-rata tertimbang.

b) Nilai indeks pelayanan

Untuk mendapatklan nilai indeks unit pelayanan, dengan cara menjumlahkan 14 unsur dari nilai rata-rata tertimbang.

### **3.6 Pengujian Kualitas Data**

Data pendapat masyarakat yang telah dimasukkan dalam masing-masing kuesioner, disusun dengan mengkompilasikan data responden yang dihimpun berdasarkan kelompok umur, jenis kelamin, pendidikan terakhir dan pekerjaan utama. Informasi ini dapat digunakan untuk mengetahui profil responden dan kecenderungan jawaban yang diberikan, sebagai bahan analisis obyektivitas.

### **3.7 Laporan Hasil Penyusunan Indeks**

Hasil akhir kegiatan penyusunan indeks kepuasan masyarakat dari setiap unit pelayanan instansi pemerintah, disusun dengan materi utama sebagai berikut:

#### **3.7.1 Indeks per unsur pelayanan**

Berdasarkan hasil penghitungan indeks kepuasan masyarakat, jumlah nilai dari setiap unit pelayanan diperoleh dari jumlah nilai rata-rata setiap unsur pelayanan.

Sedangkan nilai indeks komposit (gabungan) untuk setiap unit pelayanan, merupakan jumlah nilai rata-rata dari setiap unsur pelayanan dikalikan dengan penimbang yang sama, yaitu 0,071.

Contoh:

Apabila diketahui nilai rata-rata unsur dari masing-masing unit pelayanan adalah sebagaimana tabel berikut :

NO	UNSUR PELAYANAN	NILAI UNSUR PELAYANAN
1.	Prosedur pelayanan	3,45
2.	Persyaratan pelayanan	2,65
3.	Kejelasan petugas pelayanan	3,53
4.	Kedisiplinan petugas pelayanan	2,31
5.	Tanggung jawab petugas pelayanan	1,55
6.	Kemampuan petugas pelayanan	3,12
7.	Kecepatan pelayanan	2,13
8.	Keadilan mendapatkan pelayanan	2,43
9.	Kesopanan dan keramahan petugas	3,21
10.	Kewajaran biaya pelayanan	1,45
11.	Kepastian biaya pelayanan	1,93
12.	Kepastian jadwal pelayanan	2,31
13.	Kenyamanan lingkungan	3,03
14.	Keamanan pelayanan	1,56

maka untuk mengetahui nilai indeks unit pelayanan dihitung dengan cara sebagai berikut:

$$(3,45 \times 0,071) + (2,65 \times 0,071) + (3,53 \times 0,071) + (2,31 \times 0,071)$$

$$(1,55 \times 0,071) + (3,12 \times 0,071) + (2,13 \times 0,071) + (2,43 \times 0,071)$$

$$(3,21 \times 0,071) + (1,45 \times 0,071) + (1,93 \times 0,071) + (2,31 \times 0,071)$$

$$(3,03 \times 0,071) + (1,56 \times 0,071) = \text{Nilai indeks adalah } \mathbf{2,462}$$

Dengan demikian nilai indeks unit pelayanan hasilnya dapat disimpulkan sebagai berikut:

- a. Nilai IKM setelah dikonversi = Nilai Indeks x Nilai Dasar =  $2,462 \times 25 = 61,55$
- b. Mutu pelayanan C.
- c. Kinerja unit pelayanan Kurang Baik.

### 3.7.2 Prioritas peningkatan kualitas pelayanan

Dalam peningkatan kualitas pelayanan, diprioritaskan pada unsur yang mempunyai nilai paling rendah, sedangkan unsur yang mempunyai nilai cukup tinggi harus tetap dipertahankan.

NO	UNSUR PELAYANAN	NILAI UNSUR PELAYANAN
1.	Prosedur pelayanan	3,45
2.	Persyaratan pelayanan	2,65
3.	Kejelasan petugas pelayanan	3,53
4.	Kedisiplinan petugas pelayanan	2,31
5.	Tanggung jawab petugas pelayanan	1,55
6.	Kemampuan petugas pelayanan	3,12
7.	Kecepatan pelayanan	2,13
8.	Keadilan mendapatkan pelayanan	2,43
9.	Kesopanan dan keramahan petugas	3,21
10.	Kewajaran biaya pelayanan	1,45
11.	Kepastian biaya pelayanan	1,93
12.	Kepastian jadwal pelayanan	2,31
13.	Kenyamanan lingkungan	3,03
14.	Keamanan pelayanan	1,56