

**DISERTAI CD
PROGRAM**

PEMROGRAMAN DELPHI MENGGUNAKAN WEBCAM

 **Fadlisyah**

Penulis Beberapa buku komputer

 **FAHMI S.T, M.Sc**

Alumnus Fachhochschule Karlsruhe, Germany

 **Dayat Kurniawan**

Peneliti di Lembaga Ilmu Pengetahuan Indonesia (LIPI)



**Instalasi DSPACK - PEMROGRAMAN DATABASE
 MENGGUNAKAN WEBCAM - HISTOGRAM CITRA WEBCAM
 VIDEO WEBCAM - STREAMING VIDEO via WEBSITE**

PENERBIT

DAFTAR ISI

KATA PENGANTAR

DAFTAR ISI

BAB 1 – PENDAHULUAN	1
BAB 2 – PEMROGRAMAN DATABASE	
MENGGUNAKAN WEBCAM	15
BAB 3 – HISTOGRAM CITRA WEBCAM	31
BAB 4 – MEMBUAT VIDEO WEBCAM	45
BAB 5 – STREAMING VIDEO via WEBSITE	59

LAMPIRAN

DAFTAR PUSTAKA

BIODATA

KATA PENGANTAR

Pencerahan telah hadir menyudahi kebingungan yang bertahun-tahun telah setia menemani penulis. Dengan kehadiran buku ini telah terlepas sedikit beban penulis untuk mempublikasikan secara meluas bagaimana mempraktekkan pemrograman visual dengan input-inputnya citra dan video melalui sebuah perangkat yang tidak begitu mahal yang disebut webcam atau web camera. Dengan modal berani bertanya akhirnya penulis menemukan seorang teman konsultasi terbaik mengenai pemrograman Delphi menggunakan webcam melalui situs <http://awanday.wordpress.com>. Tidak terlalu berlebihan saya mengucapkan terima kasih kepada Mas Dayat Kurniawan yang telah memberikan banyak penjelasan dan bahkan mengirimkan beberapa sampel program ke penulis, hingga penulis mampu menyelesaikan buku ringkas ini.

Buku ini memulai pembahasannya dari penginstalasian DSPack, hingga diakhiri dengan pemrograman perekaman video melalui webcam, dan dari materi yang dipaparkan, buku ini hanya pas ditujukan kepada para pembaca yang terlebih dahulu memiliki kemampuan pemrograman yang menengah ke atas, khususnya pemrograman menggunakan Delphi. Seluruh program yang terdapat di dalam buku dapat dikonsultasikan kembali dengan penulis melalui email Fadlivision@yahoo.co.id. Sebagai penutup, penulis hanya mengucapkan selamat membaca dan semoga bermanfaat.

TERIMA KASIH KEPADA :

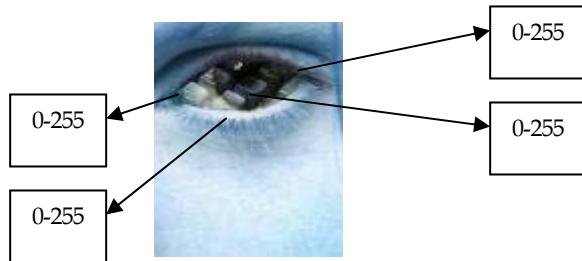
Bapak Prof. Dr. Herman Mawengkang., Prof. Dr. Opim Salim Sitompul., Dr. Rila Mandala., Prof. Dr. Utari Sumarmo., (alm) Prof. Dr. Dedi Supriadi., Prof. Drs. A. Hadi Arifin, M.Si., Bapak M. Arhami, M.Kom., Bapak Dr. Setiawan Hadi (UNPAD), Bapak Ir. Syamsul Bahri, M.Si., **Lettu Inf. Ismail Mangki.**, Bapak Dr. Nasir Usman., M.Pd., Ibu Dr. Murniati., M.Pd., Bapak Nasir Ismail Mangki., Ibunda Nezi Akhtar., Bapak Apridar, SE, M.Si., Nurlaila, M.Pd., Iskandarsyah., Muhammad Iqbal., Lisa Pebrina., Mirza Zamzami., Muchsin., Pak A. Razaq, M.Pd., dan seluruh rekan-rekan yang telah membantu secara moril kepada penulis untuk terus merealisasikan buku ini.

Bab 1

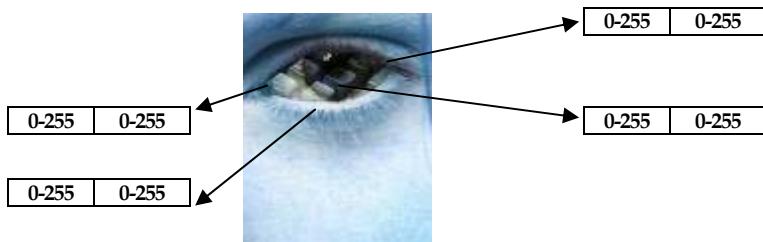
PENDAHULUAN

1.1 Format Citra

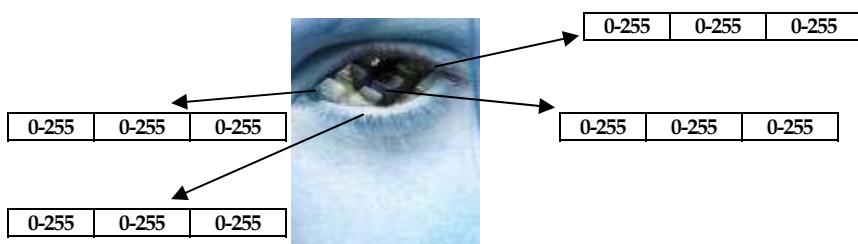
Citra direpresentasikan oleh matriks data yang memuat berbagai informasi tentang nilai fungsi citra tersebut. Dengan kata lain, citra yang tampak oleh mata kita sebenarnya merupakan kumpulan nilai-nilai tertentu yang membentuk suatu pola berdasarkan keadaan yang telah dikondisikan. Berbagai tingkatan derajat keabuan yang lazim digunakan adalah 1 bit, 4 bit, 8 bit, 16 bit, 24 bit, 32 bit dengan representasi nilai-nilai intensitas yang berbeda untuk masing-masing format pixel yang dikandungnya. Misalkan, untuk derajat keabuan 8 bit, maka citra akan memiliki rentangan nilai intensitas dari 0 hingga 255 (2^8). Untuk citra 16 bit, maka rentangan nilai-nilainya adalah 2×8 bit yaitu dari 000-000 hingga 255-255, dan begitu juga untuk citra 24 bit akan memiliki rentangan nilai intensitasnya 3×8 bit dari 000-000-000 hingga 255-255-255. Dengan kata lain, satu pixel yang terdapat pada citra 8 bit akan direpresentasikan oleh layer yang memiliki nilai 0-255 pada tabel citra, sedangkan untuk citra 24 bit, satu pixel akan direpresentasikan oleh 3 layer sekaligus, di mana setiap kotaknya akan memiliki jangkauan nilai 0-255. Untuk memperjelas maksud dari representasi pixel pada citra ke dalam kotak tabel citra, maka dapat kita ilustrasikan seperti gambar 1.1, gambar 1.2, dan gambar 1.3.



Gambar 1.1 Satu titik atau pixel pada citra 8 bit diwakili oleh suatu nilai antara 0 – 255.



Gambar 1.2 Satu titik atau pixel pada citra 16 bit diwakili oleh dua layer nilai antara 0 – 255.



Gambar 1.3 Satu titik atau pixel pada citra 24 bit diwakili oleh tiga layer nilai antara 0 – 255.

Jika ketiga layer pada tabel citra 24 bit memiliki nilai-nilai yang sama, maka pixel yang dihasilkan akan memiliki sifat keabu-abuan, ingat pembahasan tentang pengkonversian citra 24 bit menjadi citra grayscale.

Format citra yang akan digunakan di dalam buku ini adalah 24 bit, yang merupakan standar citra yang dihasilkan dari peng-capture-an menggunakan webcam penulis.

1.2 Instalasi DSPack

Sebelum membuat program penangkapan citra melalui webcam menggunakan Delphi, maka kita harus terlebih dahulu menginstalasi DSPack. DSPack yang kita gunakan adalah DSPack versi 2.3.4 yang dapat didownload pada www.progdigy.com atau dengan menghubungi penulis melalui Fadlivision@yahoo.co.id.

DSPack merupakan sekumpulan komponen dan class untuk menuliskan berbagai aplikasi multimedia menggunakan MS Direct Show dan DirectX technologies. DSPack didesain untuk bekerja dengan DirectX 9 pada sistem operasi Win9X, ME, 2000, dan Windows XP. DSPack didesain juga untuk bekerja dengan Delphi 5,6,7 dan CPP Builder 6.

Langkah-langkah instalasi yang perlu dilakukan adalah :

1. Biasanya DSPack 2.3.4 yang didownload pada situs seperti alamat situs di atas, akan berupa file Zip. Copy file tersebut ke direktori Delphi, C:\Program Files\ Borland\ Delphi7\.
2. Ekstrak atau unzip file tersebut.
3. Pada direktori C:\Program Files \Borland \Delphi7\ DSPACK234 \packages,
 - Kompilasikan file DirectX 9 Package (DirectX9_Dx.dpk), dengan mengklik 2x pada file  DirectX9_D7.

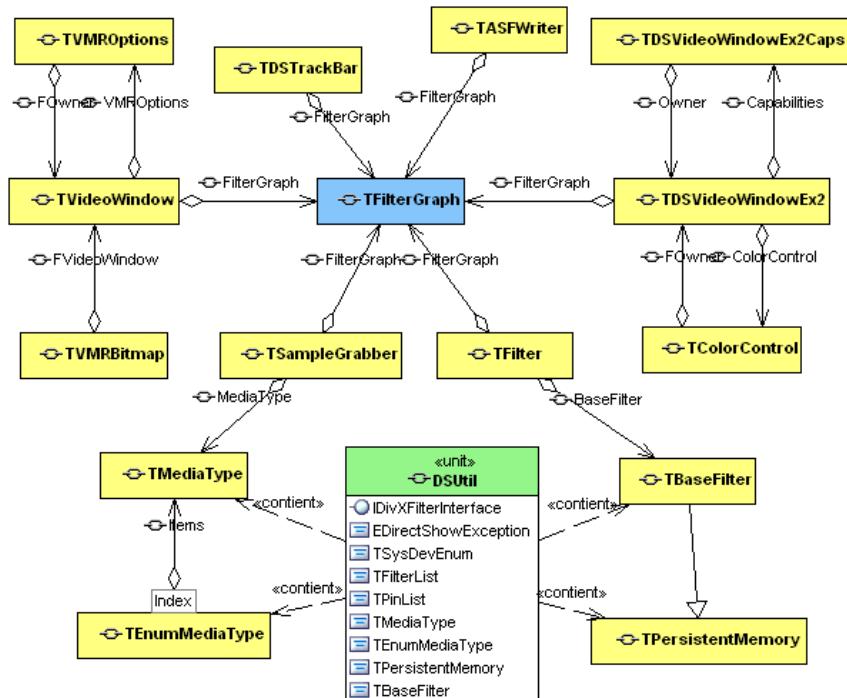
- Kompilasikan file DSPack Package (DSPack_Dx.dpk), dengan mengklik 2x pada file  DSPack_D7, dan
 - Instalasikan file Design Package (DSPackDesign_Dx.dpk), dengan mengklik 2x pada file  DSPackDesign_D7.

Setelah selesai instalasi, lalu jalankan Delphi, maka pada halaman komponen akan muncul berbagai komponen berikut :



yang terdapat pada halaman kontrol DSPack.

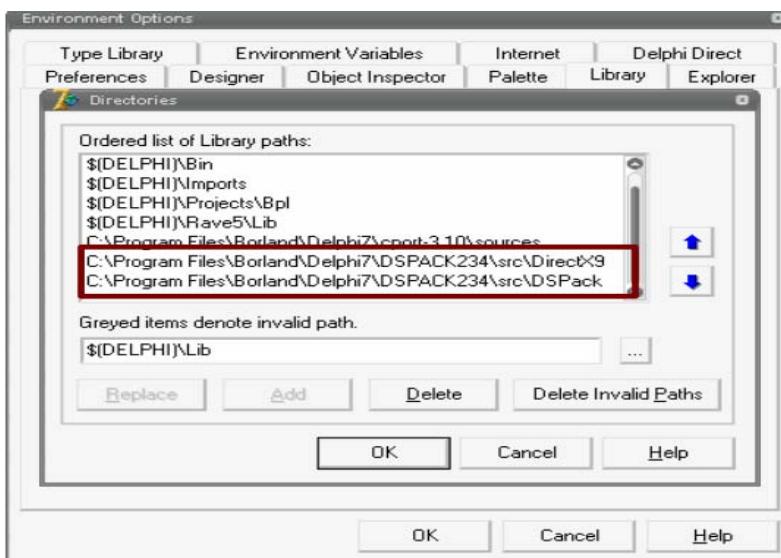
Hubungan antara berbagai komponen dan kelas pada DSPack diilustrasikan pada gambar 1.4.



Gambar 1.4 Berbagai komponen yang terdapat pada DSPack.

1.3 Permasalahan yang Sering Terjadi Dalam Instalasi DSPack

Sering terjadi di dalam mengkompilasi file DSPack Package (DSPack_Dx.dpk), atau setelah mengeksekusi file  DSPack_D7, muncul pesan kesalahan [fatal error] Dspack.pas(37).file not found 'jedi.inc' dan pesan kesalahan yang sama juga muncul dalam menginstalasi file Design Package (DSPackDesign_Dx.dpk), atau setelah mengeksekusi file  DSPackDesign_D7. Pesan kesalahan tersebut muncul karena file 'jedi.inc' tidak ditemukan pada direktori C:\Program Files \Borland \Delphi7\ DSPACK234 \packages. Mengatasi kesalahan tersebut, maka cari pada direktori mana letak file 'jedi.inc' dan setelah ditemukan copy file 'jedi.inc' ke direktori C:\Program Files \Borland \Delphi7\ DSPACK234 \packages. Hal yang sama juga berlaku untuk beberapa file yang lain yang tidak ditemukan baik dalam proses kompilasi, instalasi, maupun eksekusi. Cara lain mengatasi kesalahan di atas adalah dengan mengatur halaman library, melalui menu Tools → Environment Option → Library Path → tambahkan folder yang terdapat file 'jedi.inc' tersebut. Mengatasi masalah diatas tidak cukup hanya dengan mengkopi file jedi.inc ke folder ..\packages karena nanti pada saat run akan muncul lagi pesan kesalahan [fatal Error] unit1.pas(7): File not Found: 'DSPack.dcu'. Untuk mengatasi kedua masalah diatas adalah dengan menambahkan path src ke library seperti terlihat pada gambar di bawah ini



1.4 Program Sederhana Menangkap Citra Menggunakan Webcam

Langkah-langkah yang perlu dilakukan di dalam merancang program menangkap citra menggunakan webcam adalah sebagai berikut :

1. Jalankan Delphi.
2. Tambahkan pada Form Delphi berbagai komponen berikut : FilterGraph1 , VideoWindow1 , SampleGrabber1 , Filter1 , DBImage1 , Button1 , dan MainMenu1 .
3. Atur properti berbagai komponen di atas, sebagai :

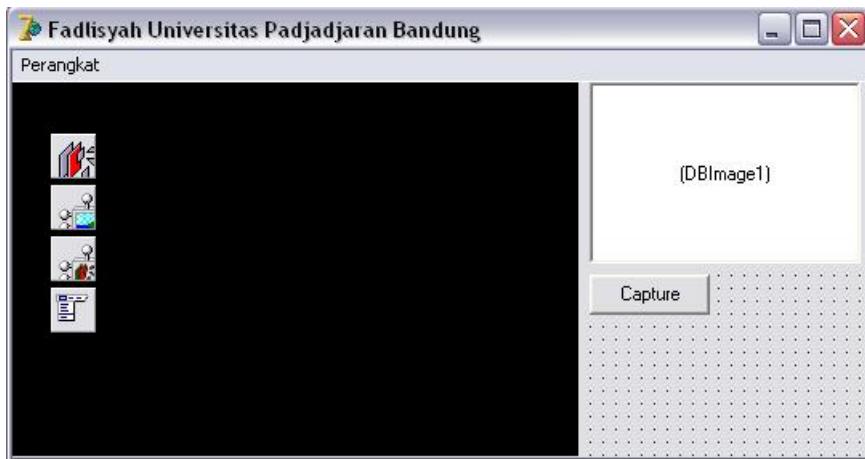
Komponen	Properti	Nilai Properti
	Mode	gmCapture
	Name	FilterGraph1

VideoWindow1	FilterGraph	FilterGraph1
	Mode	vmNormal
SampleGrabber1	FilterGraph	FilterGraph1
Filter1	FilterGraph	FilterGraph1
DBImage1	Stretch	True
Button1	Caption	Capture

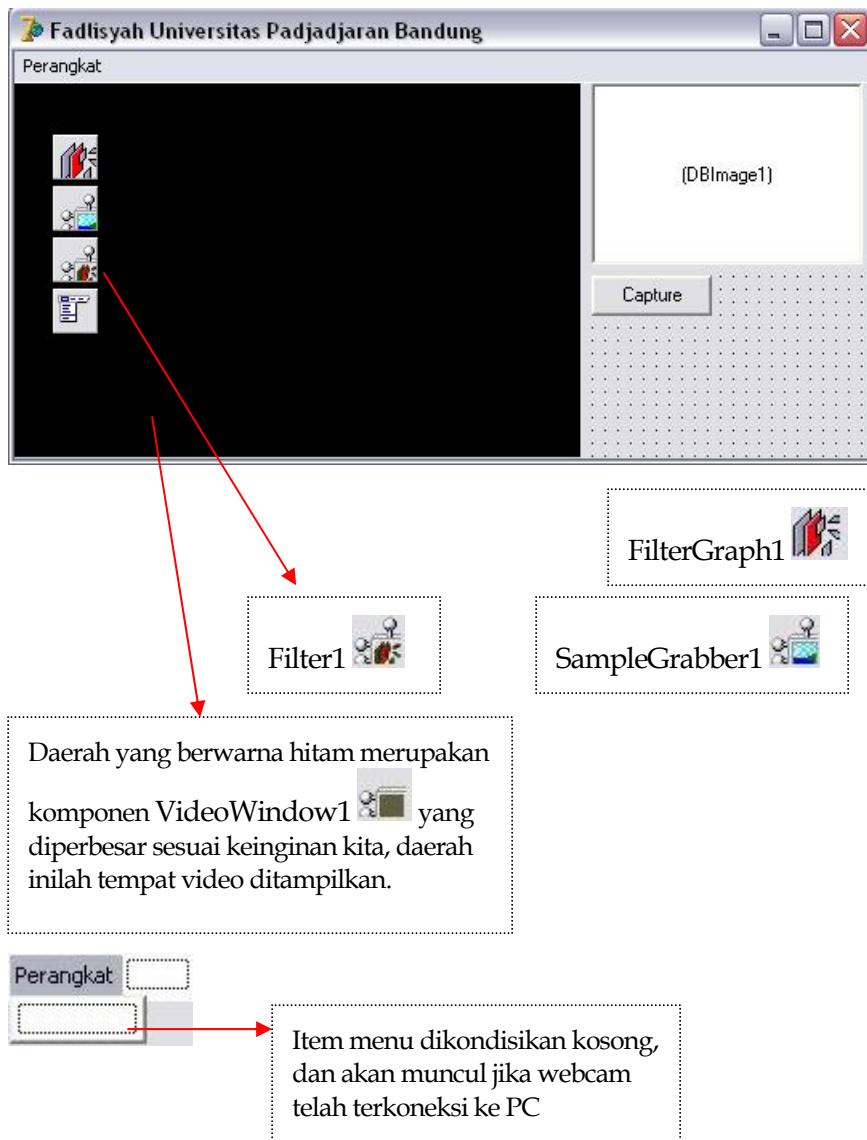
4. Klik 2x pada MainMenu1 , atur tampilan sedemikian hingga seperti :



5. Tampilan Form sekarang akan berbentuk :



Penjelasan gambar diilustrasikan berikut :



Gambar 1.5 Ilustrasi berbagai komponen yang terpasang pada Form1.

6. Klik 1x pada Form1, pada menu Events,



klik 2x pada sel OnCreate, setelah muncul halaman editor, tuliskan listing berikut :

```
procedure TForm1.FormCreate(Sender: TObject);
var
  i: integer;
  Device: TMenuItem;
begin
  SysDev:=
  TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  if SysDev.CountFilters > 0 then
    for i := 0 to SysDev.CountFilters - 1 do
    begin
      Device := TMenuItem.Create(Perangkat1);
      Device.Caption := SysDev.Filters[i].FriendlyName;
      Device.Tag := i;
      Device.OnClick := OnSelectDevice;
      Perangkat1.Add(Device);
    end;
end;
```

Ket : Kalimat yang dicetak tebal tidak perlu dituliskan, atau telah tersedia pada halaman editor.

klik 2x pada sel OnCloseQuery, setelah muncul halaman editor, tuliskan listing berikut :

```
procedure TForm1.FormCloseQuery(Sender: TObject; var
CanClose: Boolean);
begin
  SysDev.Free;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
end;
```

7. Kembali ke halaman Form1, klik 2x pada Button1, lalu tuliskan listing berikut :

```
procedure TForm1.Button1Click(Sender: TObject);
var simpan : tbitmap;
begin
  simpan:=tbitmap.create;
  SampleGrabber1.GetBitmap(DBImage1.Picture.Bitmap);
  simpan:=DBImage1.Picture.Bitmap;
  simpan.savetofile('fadli.bmp');
end;
```

8. Masih pada halaman editor, deklarasikan prosedur berikut :

```
procedure OnSelectDevice(sender: TObject);
```

di bawah,

```
private
  { Private declarations }
public
  { Public declarations }
    procedure OnSelectDevice(sender: TObject);
```

Lalu tambahkan listing untuk prosedur tersebut, pada tubuh program.

```
procedure TForm1.OnSelectDevice(sender: TObject);
begin
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
  Filter1.BaseFilter.Moniker := SysDev.GetMoniker(TMenuItem(Sender).tag);
  FilterGraph1.Active := true;
  with FilterGraph1 as ICaptureGraphBuilder2 do
```

```
    RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as IBaseFilter,
SampleGrabber1 as IBaseFilter, VideoWindow1 as IbaseFilter;
    FilterGraph1.Play;
  end;
```

Keterangan : kalimat yang diberi garis bawah (underline) merupakan kalimat yang tidak terpisah atau satu kalimat utuh.

9. Selanjutnya, tambahkan beberapa unit pada uses default sehingga menjadi :

```
uses
  Windows, Messages, SysUtils, Variants, Classes,
  Graphics, Controls, Forms, Dialogs, StdCtrls, DBCtrls,
  Menus, DirectShow9, DSPack, DSUtil, ExtCtrls;
```

10. Deklarasikan variabel berikut di bawah var header.

```
var
  Form1: TForm1;
  SysDev: TSysDevEnum;
```

11. Periksa kembali keseluruhan listing program, apakah persis seperti keseluruhan listing berikut :

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, DBCtrls, Menus, DirectShow9, DSPack, DSUtil, ExtCtrls;

type
  TForm1 = class(TForm)
    FilterGraph1: TFilterGraph;
    SampleGrabber1: TSampleGrabber;
    Filter1: TFilter;
    MainMenu1: TMainMenu;
    Perangkat1: TMenuItem;
    Button1: TButton;
    VideoWindow1: TVideoWindow;
    DBImage1: TDBImage;
    procedure Button1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);

private
```

```
{ Private declarations }
public
{ Public declarations }
procedure OnSelectDevice(sender: TObject);
end;

var
Form1: TForm1;
SysDev: TSysDevEnum;

implementation

{$R *.dfm}
//-----
procedure TForm1.Button1Click(Sender: TObject);
var simpan : tbitmap;
begin
simpan:=tbitmap.create;
SampleGrabber1.GetBitmap(DBImage1.Picture.Bitmap);
simpan:=DBImage1.Picture.Bitmap;
simpan.savetofile('fadli.bmp');
end;
//-----
procedure TForm1.FormCreate(Sender: TObject);
var
i: integer;
Device: TMenuItem;
begin
SysDev:= TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
if SysDev.CountFilters > 0 then
for i := 0 to SysDev.CountFilters - 1 do
begin
Device := TMenuItem.Create(Perangkat1);
Device.Caption := SysDev.Filters[i].FriendlyName;
Device.Tag := i;
Device.OnClick := OnSelectDevice;
Perangkat1.Add(Device);
end;
end;
//-----
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
begin
SysDev.Free;
FilterGraph1.ClearGraph;
FilterGraph1.Active := false;
end;
//-----
```

```
procedure TForm1.OnSelectDevice(sender: TObject);
begin
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
  Filter1.BaseFilter.Moniker := SysDev.GetMoniker(TMenuItem(Sender).tag);
  FilterGraph1.Active := true;
  with FilterGraph1 as ICaptureGraphBuilder2 do
    RenderStream(@PIN CATEGORY_PREVIEW, nil, Filter1 as IBaseFilter,
    SampleGrabber1 as IBaseFilter, VideoWindow1 as IbaseFilter);
  FilterGraph1.Play;
end;
end.
```

12. Eksekusi program (F9).

Salah satu tampilan program setelah eksekusi,



Bab 2

PEMROGRAMAN DATABASE MENGGUNAKAN WEBCAM

2.1 Membangun Tabel

Sebelum kita merancang program database yang menerima berbagai input data-datanya dari perangkat webcam, maka terlebih dahulu kita lakukan langkah-langkah berikut :

1. Buka Database Desktop.
2. Atur Working Directory ke folder baru yang anda buat, dan selanjutnya tentukan alias database anda melalui menu Tools → Alias Manager.
3. Buat struktur tabel seperti berikut :

	Field Name	Type	Size	Key
1	Nim	A	9	*
2	Nama	A	10	
3	Foto	G		

4. Simpan tabel dengan nama tabel yang anda tentukan, disarankan tempat penyimpanan tabel berada satu lokasi dengan folder yang anda buat untuk penyimpanan proyek program.

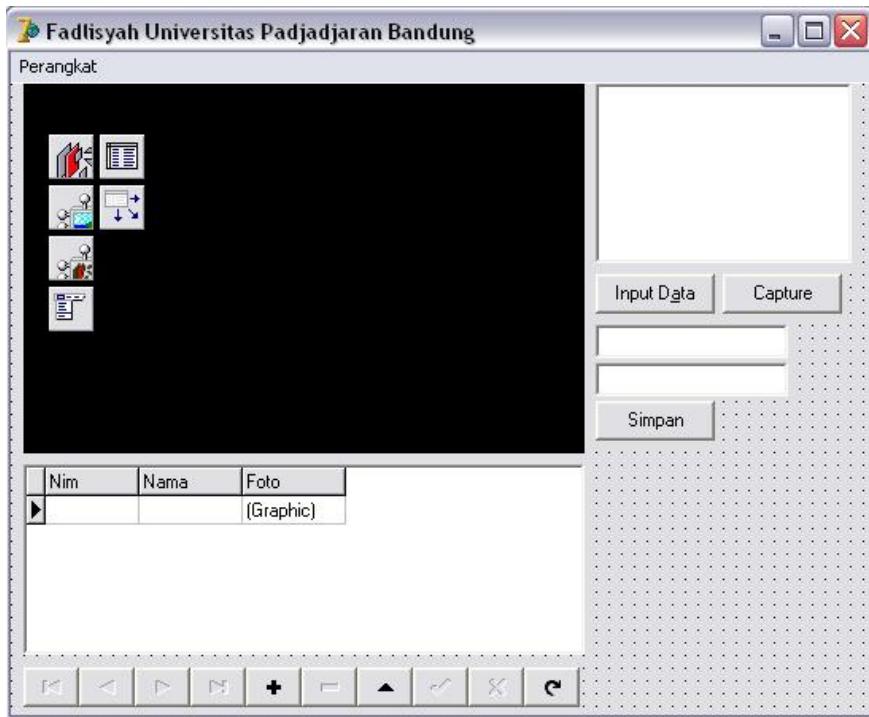
2.2 Mengkoneksikan Webcam ke Tabel Melalui Proyek Delphi

1. Buka kembali proyek Delphi pada Bab 1,

2. Tambahkan icon Table1 , DataSource1 , DBGrid1 , DBNavigator1 , DBEdit1 , DBEdit2 , Button2 , dan Button3 .
3. Atur properti berbagai komponen di atas, sebagai :

Komponen	Properti	Nilai Properti
 Table1	DatabaseName	DBDEMONS
	TableName	WebCam.db
	Active	True
 DataSource1	Enabled	True
	DataSet	Table1
 DBGrid1	DataSource	DataSource1
 DBNavigator1	DataSource	DataSource1
 Button2	Caption	Input Data
 Button3	Caption	Simpan
	Visible	False
 DBEdit1	DataSource	DataSource1
	DataField	Nim
	Visible	False
 DBEdit2	DataSource	DataSource1
	DataField	Nama
	Visible	False
 DBImage1	DataSource	DataSource1
	DataField	Foto
	Visible	False

4. Setelah berbagai komponen di atas ditambahkan pada Form1, dan juga telah diatur berbagai propertinya, maka tampilan Form yang diharapkan akan berupa :



5. Klik 2x pada Button2 atau Tombol Input Data, lalu tuliskan listing berikut :

```
Table1.append;  
button3.Visible:=true;  
dbedit1.Visible:=true;  
dbedit2.Visible:=true;  
dbimage1.Visible:=true;
```

6. Klik 2x pada Button3 atau Tombol Simpan, lalu tambahkan listing berikut :

```
button1.Click;
```

```

table1.Post;
button1.Visible:=false;
button3.Visible:=false;
dedit1.Visible:=false;
dedit2.Visible:=false;

```

7. Eksekusi program dengan menekan tombol F9.

2.3 Mengatur Laporan Melalui Fasilitas Rave

Untuk mengatur laporan, maka masih pada Form yang sama,

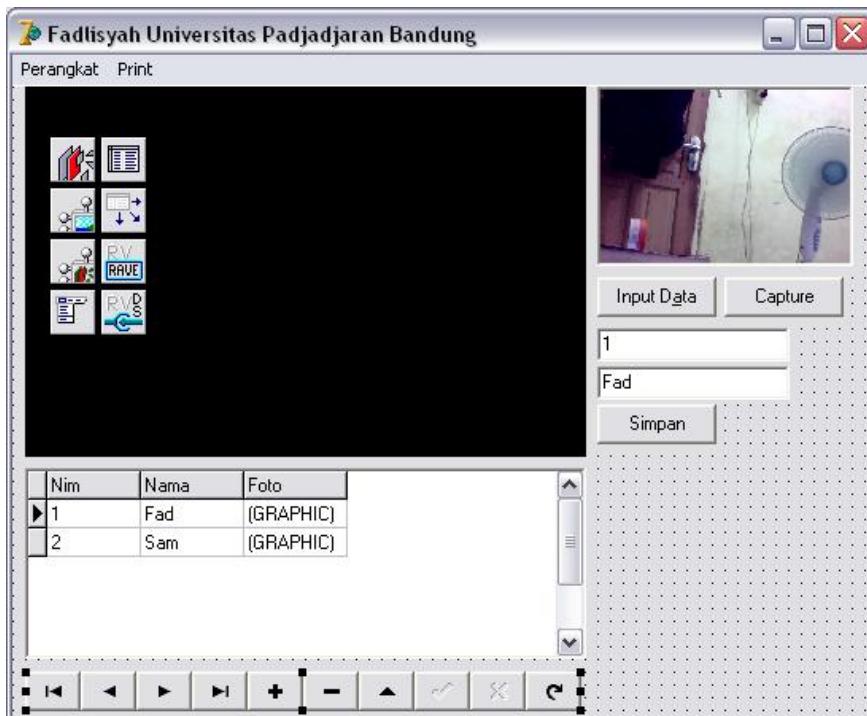
tambahkan komponen RvProject1 , dan RvDataSetConnection1 . Kedua komponen tersebut dapat di temukan pada page control Rave.



Atur properti berbagai komponen yang baru ditambah di atas, sebagai berikut :

Komponen	Properti	Nilai Properti
RvProject1 	ProjectFile	Belum bisa ditentukan
RvDataSetConnection1 	dataSet	Table1

Perhatian : properti RvProject1 , masih belum bisa diatur propertinya sebelum kita merancang laporan menggunakan Rave Reports. Jadi tampilan Form setelah ditambah komponen RvProject1 , dan RvDataSetConnection1  adalah :

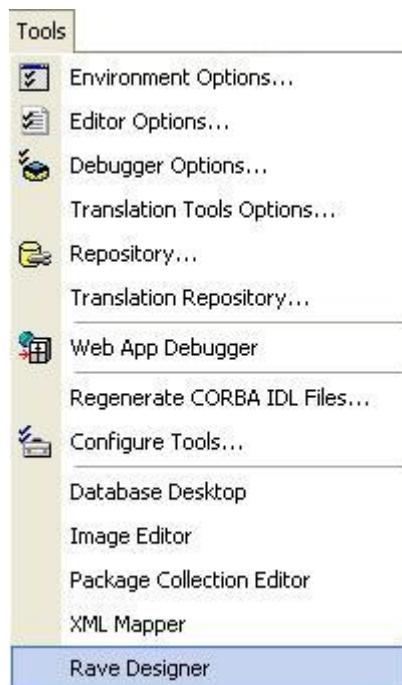


Perhatikan, pada tampilan di atas terdapat item menu Print, tambahkan item menu print seperti tampilan berikut :

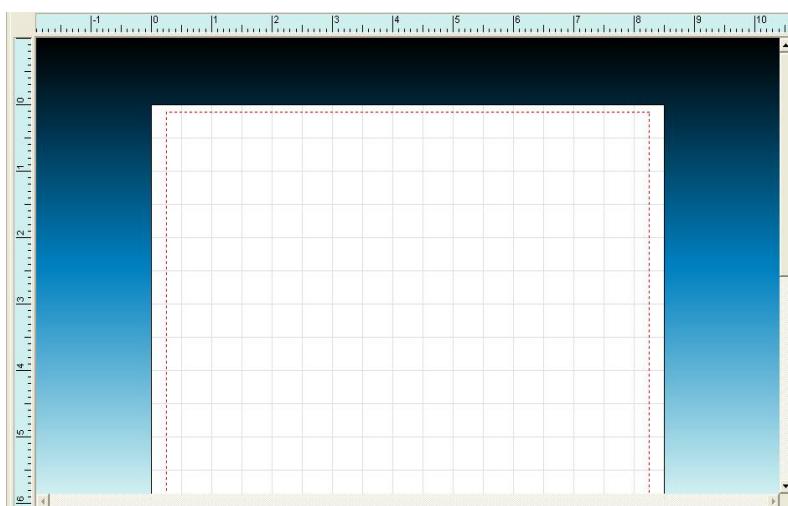


Selanjutnya untuk mendesain laporan data yang terdapat pada tabel, ikuti langkah-langkah berikut :

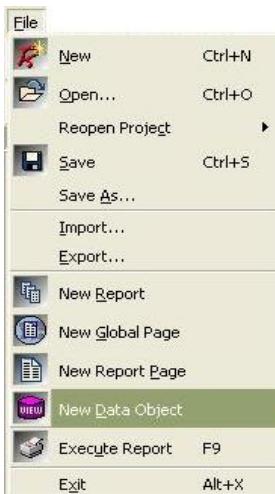
1. Jalankan Rave Designer.



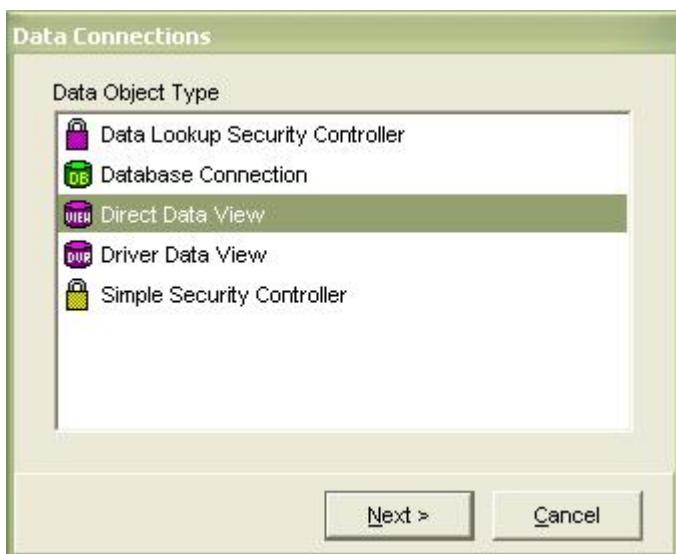
Tidak lama kemudian muncul halaman Rave Project.



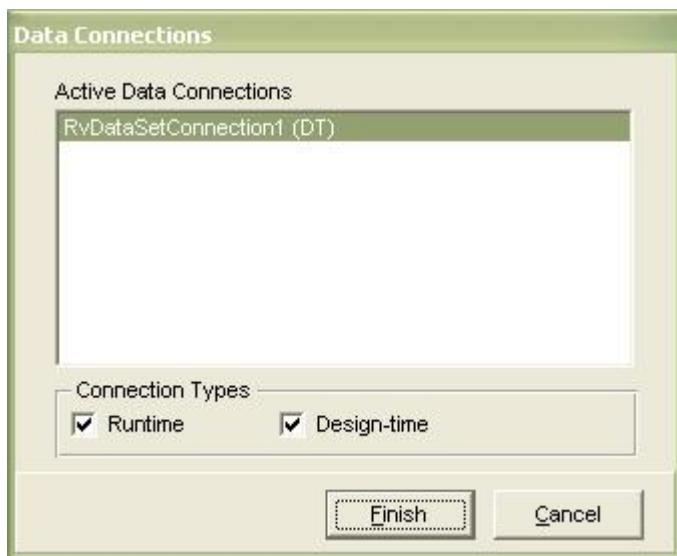
2. Lalu sorot pada MainMenu File,



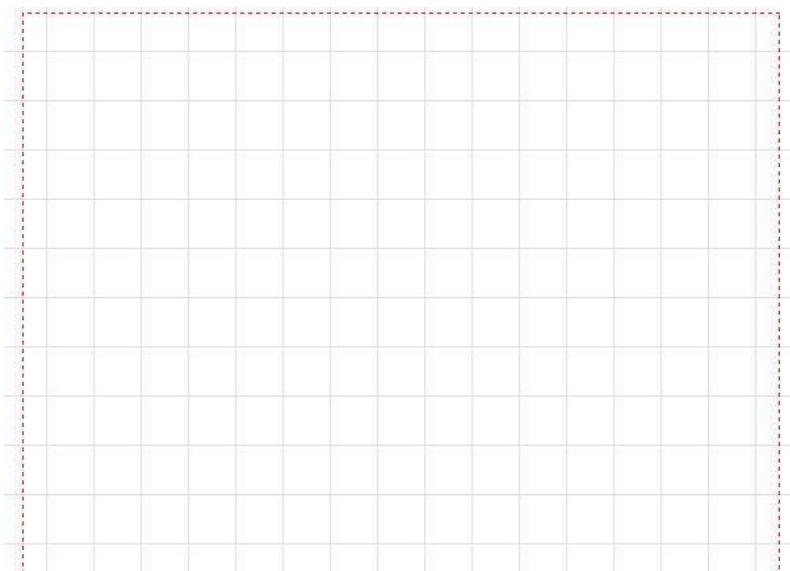
pilih New Data Object . Maka akan muncul tampilan berikut :



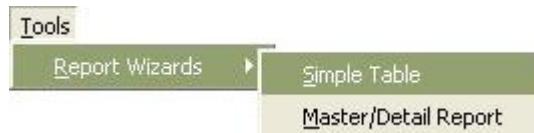
3. Pilih Direct Data View dan tekan Button Next, dan muncul tampilan berikut :



4. Pilih RvDataSetConnection1 (DT) dan lalu tekan Button Finish.
5. Sekarang tampilan pada layar komputer anda adalah (tetap) :



6. Lalu sorot ke menu Tools, pilih Report Wizards, setelah muncul tampilan :



pilih Simple Table.

7. Tampilan sekarang adalah :



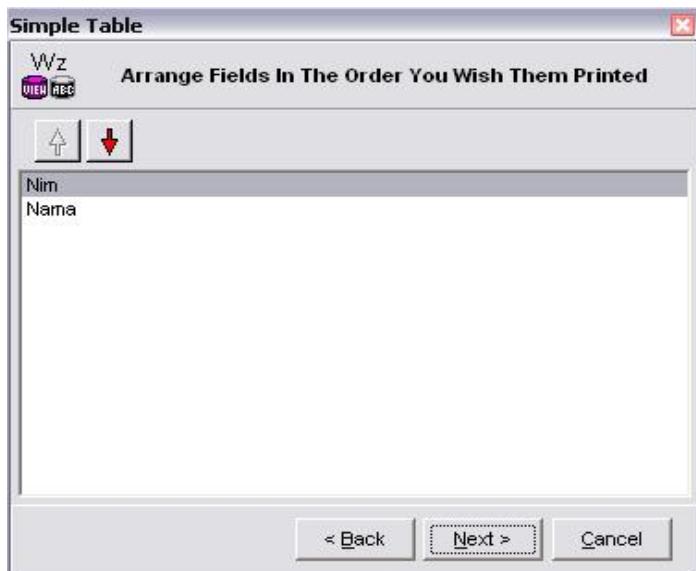
pilih DataView1. Lalu muncul tampilan berikut :



8. Klik Button All jika ingin menampilkan semua.

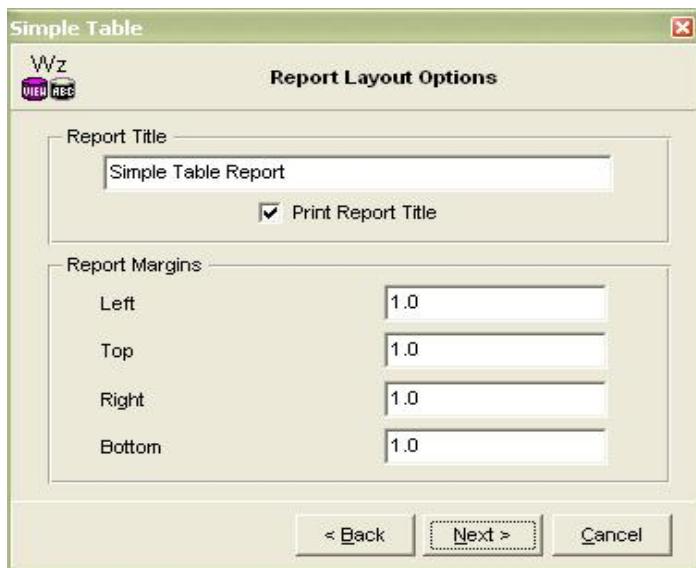
Nim
 Nama

Lalu klik Button Next>. Sekarang komputer meminta konfirmasi urutan field melalui tampilan :

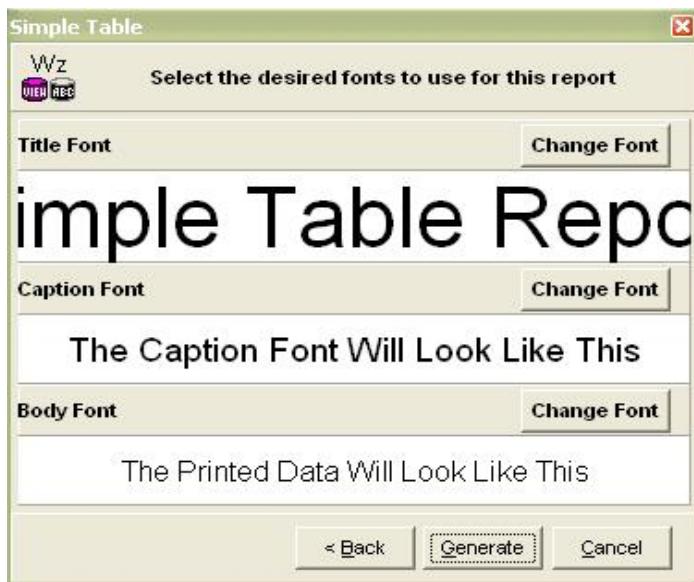


tekan Button Next>.

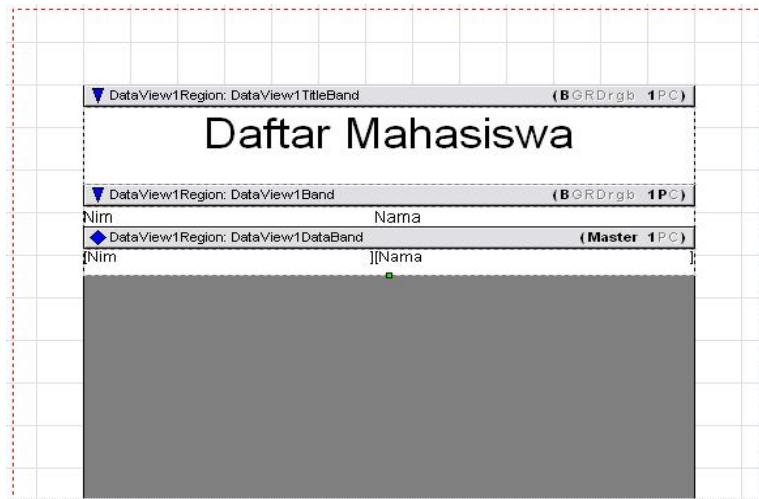
Tampilan selanjutnya adalah :



kalau tidak ada pengaturan batas-batas printout laporan, maka klik Button Next>. Muncul tampilan :

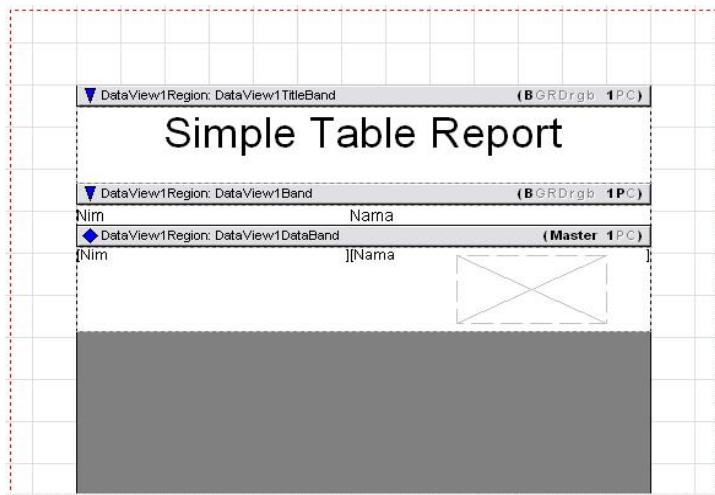


dan jika tidak ada pengaturan ukuran font yang digunakan, maka klik Button Generate. Dan bentuk laporan yang dihasilkan akan berupa tampilan berikut :



Klik pada bagian DataView1Region : DataView1DataBand, daerah yang berwarna putih di bawah bar DataView1Region : DataView1DataBand diperluas seperlunya, dan tambahkan

komponen Bitmap Component  (terdapat dalam page control Standard pada Rave Reports), lalu letakkan di dalam daerah putih di bawah DataView1Region : DataView1DataBand yang diperluas tadi, tampilan akhir akan berupa :



Atur properti Bitmap Component,

DataView	DataView1
DataField	Foto
MatchSide	msBoth

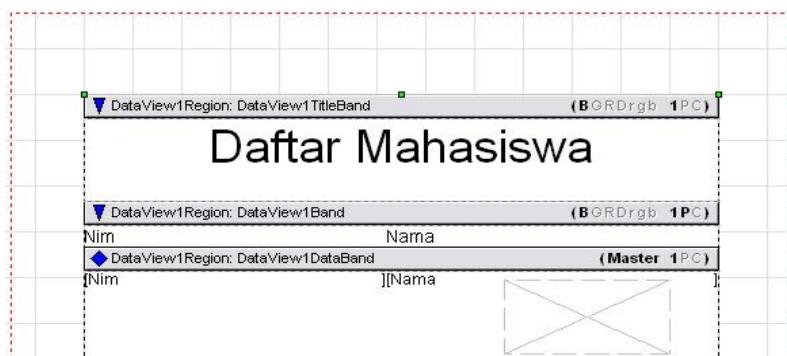
Nilai properti MatchSide adalah msBoth atau setara dengan nilai properti Stretch sama dengan True.

Untuk melihat efek penambahan Bitmap Component  , maka anda dapat melakukan eksekusi dengan menekan tombol F9.

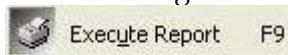
9. Klik 1x pada tulisan Simple Table Report, lalu atur properti text menjadi Daftar Mahasiswa.

TitleText: Text component	
Anchor	(Top / Left)
Color	FFFF
DevLocked	False
DisplayOn	doParent
Font	Arial,36
FontJustify	piCenter
FontMirror	
Left	0
Locked	False
Mirror	
Name	TitleText
Rotation	0
Top	0
Truncate	False
Width	6.5

Tampilan laporan sekarang adalah :



10. Eksekusi dengan menekan tombol F9 atau melalui menu File



. Muncul tampilan berikut :



dan klik ok.

Laporan akhir kita akan berbentuk :

Daftar Mahasiswa

Nim	Nama	
1	Fad	
2	Sam	

Simpan file (Project1.rav) ke folder di mana anda menyimpan file database dan project Delphi anda, misal : D:\#Data Fadlisyah\~EX Delphi \-WebCam Delphi \DBWEB \Project1.rav. Lalu keluar dari Rave Reports dan kembali ke halaman Form Delphi.

Atur kembali properti berbagai komponen berikut :

Komponen	Properti	Nilai Properti
RvProject1 	ProjectFile	D:\#Data Fadlisyah\~EX Delphi \-WebCam Delphi \DBWEB \Project1.rav
RvDataSetConnection1 	dataSet	Table1



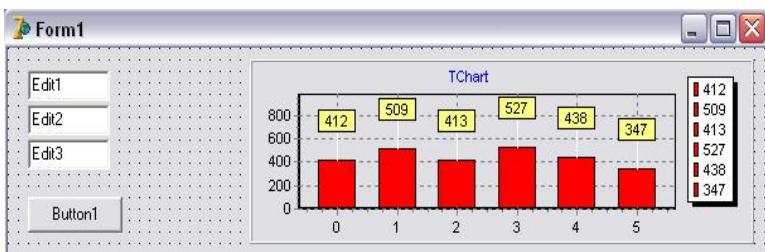
Bab 3

HISTOGRAM CITRA WEBCAM

3.1 Chart Sederhana

Berbagai kamera digital biasanya dilengkapi dengan histogram. Histogram bertujuan menampilkan distribusi nilai intensitas pada citra secara visual. Sebelum kita membahas lebih jauh mengenai histogram citra, maka terlebih dahulu kita akan mempelajari penyajian data dengan grafik secara sederhana. Untuk itu ikuti langkah-langkah berikut :

1. Jalankan Delphi.
2. Tambahkan komponen Chart1 , Edit1 , Edit2 , Edit3 , dan Button1  pada Form1.
3. Tampilan yang dimaksud akan berupa :

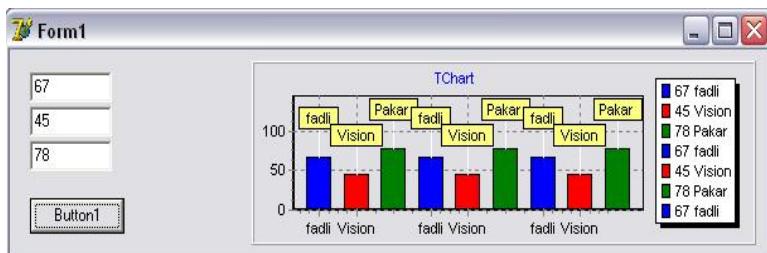


4. Klik 2x pada Chart1 , tambahkan series, dan atur jenis chart berupa jenis Bar, dan keluar dari Chart1.

5. Klik 2x pada Button1 , lalu tuliskan listing berikut :

```
series1.Add(strtoint(edit1.text), 'fadli', clblue);
series1.Add(strtoint(edit2.text), 'Vision', clred);
series1.Add(strtoint(edit3.text), 'Pakar', clgreen);
```

6. Eksekusi, dan tampilan yang dihasilkan akan seperti :



3.2 DBChart

Terlebih dahulu kita desain sebuah tabel dengan struktur berikut :

Restructure Paradox 7 Table: DB.db

Field roster:					Table properties:				
	Field Name	Type	Size	Key					
1	NIM	A	9	*					
2	Nama	A	10						
3	Nilai Angka	N							

Enter a field name up to 25 characters long.

Table properties (right side):

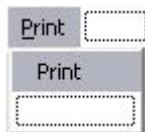
- Validity Checks:
- 1. Required Field
- 2. Minimum value:
- 3. Maximum value:
- 4. Default value:
- 5. Picture:

Buttons at the bottom:

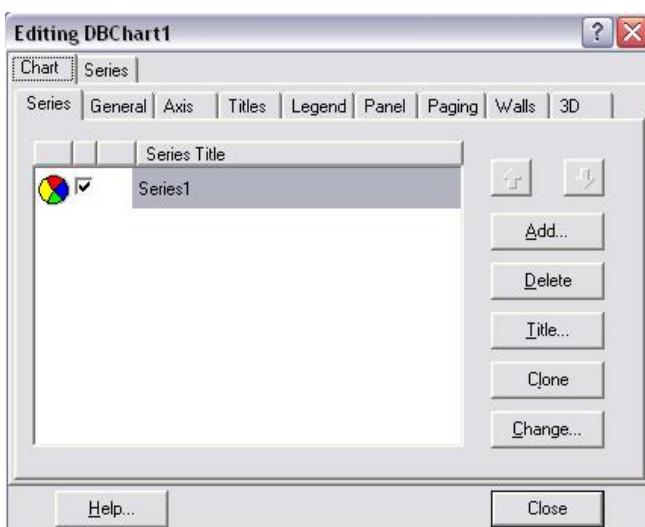
- Pack Table
-
-
-
-

selanjutnya jalankan Delphi. Tambahkan komponen Table1 , DataSource1 , DBGrid1 , DBNavigator1 , DBChart1 , dan MainMenu1  pada Form Delphi.

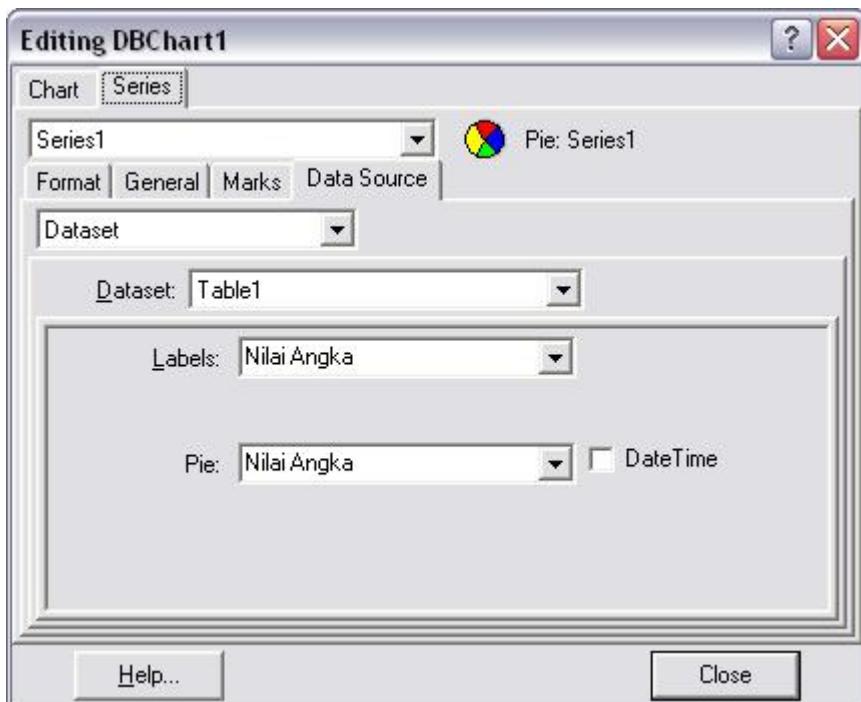
Untuk pengaturan properti Table1 , DataSource1 , DBGrid1 , DBNavigator1  tetap menggunakan pengaturan standar yang mengacu kepada tabel yang baru dibuat. Untuk MainMenu1 , klik 2x pada icon MainMenu1 , lalu set menu seperti tampilan,



dan untuk DBChart1 , klik 2x pada icon DBChart1 , tambahkan series melalui tab series, dan selesaikan dengan menentukan tipe grafik yang digunakan (untuk kasus ini : Pie).



atur sumber data DBChart1  yang digunakan, melalui halaman tab berikut :

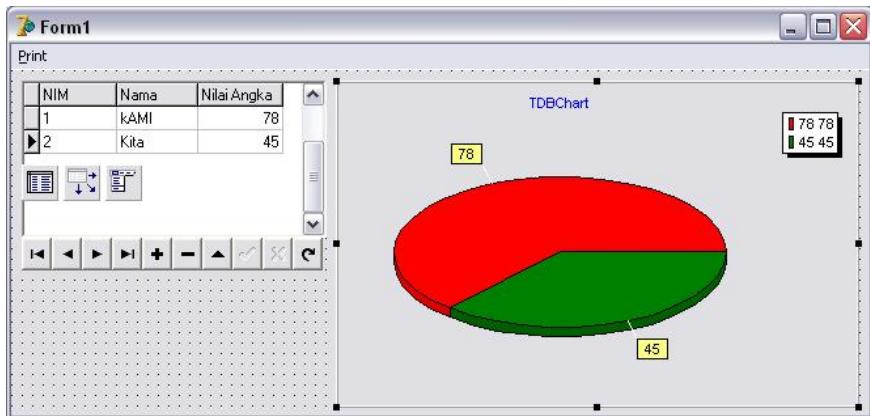


pastikan nilai properti berikut di atur merujuk ke tabel yang anda gunakan. Untuk kasus penulis, nilai propertinya adalah

DataSource	Dataset
Dataset	Table1
Pie	Nilai Angka

dan akhiri dengan menekan tombol Close.

Tampilan Form1 yang dikehendaki kira-kira akan berupa :



Untuk melihat efek dari penambahan data pada tabel terhadap tampilan grafis (DBChart1) yang dibuat, klik F9 atau eksekusi program melalui icon .

Keterangan tambahan : untuk mencetak tampilan grafis pada Form eksekusi, maka klik 2x pada item menu Print, lalu tuliskan kode berikut :

```
DBChart1.Print;
```

atau

```
DBChart1.Print Landscape;
```

Lalu eksekusi kembali proyek di atas. Lihat sendiri perbedaan kedua mode pencetakan di atas.

3.3 Histogram

Histogram citra merupakan grafik yang memuat penyebaran nilai-nilai intensitas pixel dari suatu citra. Misalkan citra memiliki L grey-level, dari 0 sampai $L - 1$ (citra 8 bit memiliki rentang 0 hingga 255 grey-level), maka histogram dikalkulasikan sebagai :

$$h_i = \frac{n_i}{n}, \quad i = 0, 1, \dots, L - 1$$

di mana n_i merupakan jumlah pixel yang memiliki grey-level i , dan n merupakan jumlah seluruh pixel.

Contoh, suatu citra memiliki nilai-nilai intensitas pixel berikut:

1	8	4	3	6	2	5	2	8	4	6	2	5
0	3	8	3	6	5	4	0	3	8	3	8	7
3	8	4	7	6	2	8	3	7	3	7	6	1
0	9	8	0	5	4	8	5	9	3	7	2	9

$$n = 52$$

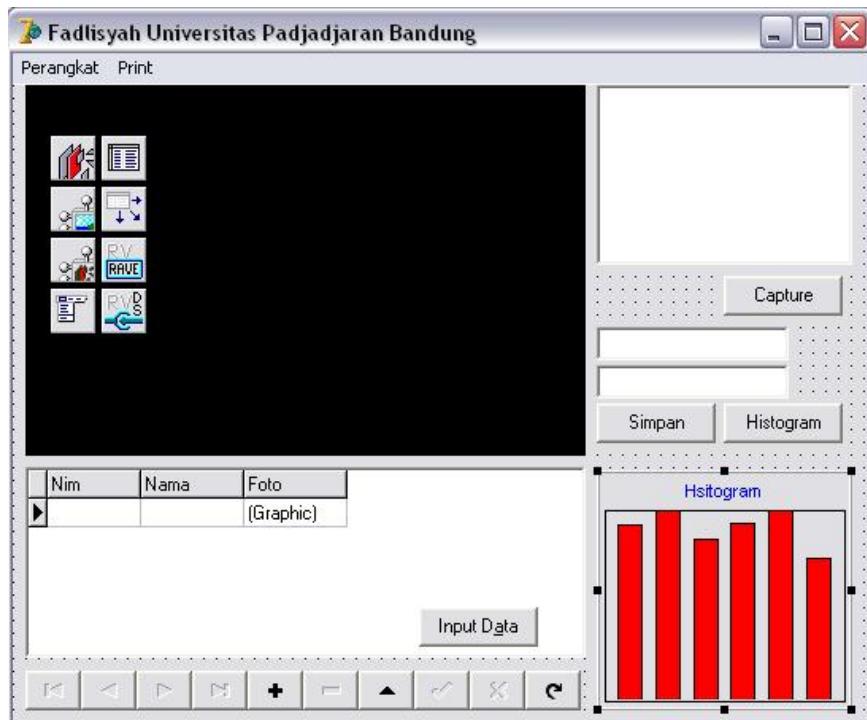
akan memberikan histogram sebagai berikut :

0	****	$\frac{n_0}{n} = \frac{4}{52} = 0,07692$
1	**	$\frac{n_1}{n} = \frac{2}{52} = 0,03846$
2	*****	$\frac{n_2}{n} = \frac{5}{52} = 0,09615$
3	*****	$\frac{n_3}{n} = \frac{9}{52} = 0,17308$
4	*****	$\frac{n_4}{n} = \frac{5}{52} = 0,09615$
5	*****	$\frac{n_5}{n} = \frac{5}{52} = 0,09615$
6	*****	$\frac{n_6}{n} = \frac{5}{52} = 0,09615$
7	*****	$\frac{n_7}{n} = \frac{5}{52} = 0,09615$
8	*****	$\frac{n_8}{n} = \frac{9}{52} = 0,17308$
9	***	$\frac{n_9}{n} = \frac{3}{52} = 0,05769$

3.4 Pemrograman Histogram Berdasarkan Citra yang Di-Capture Menggunakan Webcam

Untuk membuat program histogram, kita dapat menggunakan program pada bab 2, program yang menerima input citranya melalui webcam. Buka kembali program yang terdapat pada bab 2, tambahkan

komponen Chart1 , dan Button4  pada Form1 seperti tampilan Form di bawah.



Klik 2x pada icon Chart, atur berbagai properti berikut :

Chart		
	Series	1. Klik Button [Add], hilangkan tanda √ pada checkbox 3D, lalu pilih Bar, klik

		Button [Ok].
	Axis	2. Hilangkan tanda <input checked="" type="checkbox"/> pada checkbox Show Axis.
	Titles	3. Beri judul HISTOGRAM.
	Legend	4. Pastikan tanda <input checked="" type="checkbox"/> pada checkbox Visible tidak ada.
	Panel	5. Klik RadioButton Raised pada RadioGroup BevelInner. 6. Klik RadioButton Lowered pada RadioGroup BevelOuter.
Series	Marks	7. Pastikan tanda <input checked="" type="checkbox"/> pada checkbox Visible tidak ada

Atur caption Button4  menjadi &Histogram. Klik 2x pada Button4  atau &Histogram, setelah muncul halaman editor, tuliskan kode berikut :

```
procedure TForm1.Button4Click(Sender: TObject);
var i : Integer;
begin
    series1.clear;
    Histogram;
    for i := 0 to 255 do
        Series1.AddXY(i,datamod[i],'-',clblack);
end;
```

setelah selesai, maka tambahkan listing untuk prosedur histogram.

```
procedure TForm1.Histogram;
var temp      : pbytearray;
    x,y,jum : integer;
    tengah   : integer;
    a,b,c   : real;
begin
    jum:=0;
```

```
for y:=0 to (simpan.Height-1) do
begin
  temp := simpan.ScanLine[y];
  x:=0;
  repeat
    a := 0.11*temp[x];
    b := 0.59*temp[x+1];
    c := 0.3*temp[x+2];
    tengah := round(a+b+c);
    datamod[tengah] := datamod[tengah]+1;
    inc(jum);
    inc(x,3);
  until x>3*(simpan.Width-1);
end;
for x:=0 to 255 do
  datamod[x]:=datamod[x]/jum;
end;
```

Periksa kembali listing yang anda buat, apakah sudah menyerupai listing berikut :

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms, Dialogs, StdCtrls, DBCtrls, Menus,
  DirectShow9, DSPack, DSUtil, ExtCtrls, Grids, DBGirds, DB,
  DBTables, Mask, RpCon, RpConDS, RpDefine, RpRave,
  TeeProcs, TeEngine, Chart, Series;

type
  TForm1 = class(TForm)
    FilterGraph1: TFilterGraph;
    SampleGrabber1: TSampleGrabber;
    Filter1: TFilter;
    MainMenul: TMainMenu;
    Perangkat1: TMenuItem;
    Button1: TButton;
    VideoWindow1: TVideoWindow;
    DBImage1: TDBImage;
    DataSourcel: TDataSource;
    Table1: TTable;
    DBGrid1: TDBGrid;
```

```
DBEdit1: TDBEdit;
DBEdit2: TDBEdit;
Button2: TButton;
DBNavigator1: TDBNavigator;
Button3: TButton;
RvProject1: TRvProject;
RvDataSetConnection1: TRvDataSetConnection;
Print1: TMenuItem;
Print2: TMenuItem;
Chart1: TChart;
Button4: TButton;
Series1: TBarSeries;
procedure Button1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure FormCloseQuery(Sender: TObject; var CanClose:
  Boolean);
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Print2Click(Sender: TObject);
procedure Button4Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
  procedure OnSelectDevice(sender: TObject);
  procedure Histogram;
end;

var
  Form1: TForm1;
  SysDev: TSysDevEnum;
  datamod    : array[0..255] of real;
  simpan : tbitmap;
implementation

{$R *.dfm}
procedure TForm1.Button1Click(Sender: TObject);
begin
  simpan:=tbitmap.create;
  SampleGrabber1.GetBitmap(DBImage1.Picture.Bitmap);
  simpan:=DBImage1.Picture.Bitmap;
  simpan.savetofile('fadli.bmp');
end;
procedure TForm1.FormCreate(Sender: TObject);
var
  i: integer;
```

```
Device: TMenuItem;
begin
  SysDev:= TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  if SysDev.CountFilters > 0 then
    for i := 0 to SysDev.CountFilters - 1 do
      begin
        Device := TMenuItem.Create(Perangkat1);
        Device.Caption := SysDev.Filters[i].FriendlyName;
        Device.Tag := i;
        Device.OnClick := OnSelectDevice;
        Perangkat1.Add(Device);
      end;
end;
procedure TForm1.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
begin
  SysDev.Free;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
end;
procedure TForm1.OnSelectDevice(sender: TObject);
begin
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
  Filter1.BaseFilter.Moniker := SysDev.GetMoniker (TmenuItem
  (Sender).tag);
  FilterGraph1.Active := true;
  with FilterGraph1 as ICaptureGraphBuilder2 do
    RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as
    IBaseFilter, SampleGrabber1 as IBaseFilter, VideoWindow1 as
    IbaseFilter);
  FilterGraph1.Play;
end;
procedure TForm1.Button2Click(Sender: TObject);
begin
  Table1.append;
  button3.Visible:=true;
  dbedit1.Visible:=true;
  dbedit2.Visible:=true;
  dbimage1.Visible:=true;
end;

procedure TForm1.Button3Click(Sender: TObject);
begin
  button1.Click;
  table1.Post;
  button1.Visible:=false;
  button3.Visible:=false;
```

```
dbedit1.Visible:=false;
dbedit2.Visible:=false;
end;

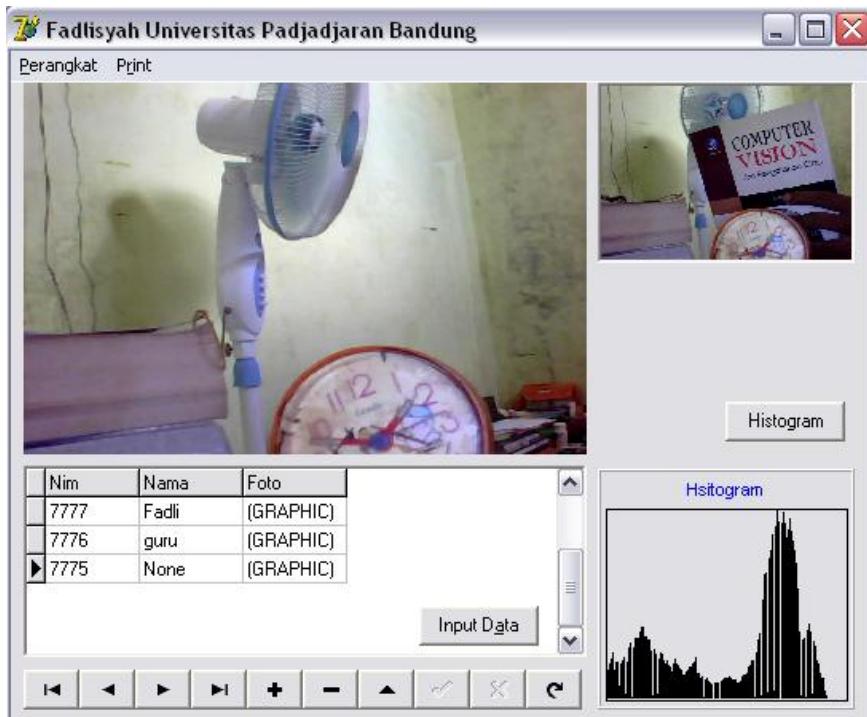
procedure TForm1.Print2Click(Sender: TObject);
begin
rvproject1.Execute;
end;

procedure TForm1.Button4Click(Sender: TObject);
var i : Integer;
begin
    series1.clear;
    Histogram;
    for i := 0 to 255 do
        Series1.AddXY(i,datamod[i],'-',clblack);
end;

procedure TForm1.Histogram;
var temp      : pbytearray;
    x,y,jum : integer;
    tengah   : integer;
    a,b,c    : real;
begin
    jum:=0;

    for y:=0 to (simpan.Height-1) do
        begin
            temp := simpan.ScanLine[y];
            x:=0;
            repeat
                a := 0.11*temp[x];
                b := 0.59*temp[x+1];
                c := 0.3*temp[x+2];
                tengah := round(a+b+c);
                datamod[tengah] := datamod[tengah]+1;
                inc(jum);
                inc(x,3);
            until x>3*(simpan.Width-1);
        end;
    for x:=0 to 255 do
        datamod[x]:=datamod[x]/jum;
end;
end.
```

Setelah dilakukan pemeriksaan ulang, maka proyek sudah siap dieksekusi. Gambar di bawah ini merupakan salah satu tampilan eksekusi.



Bab 4

MEMBUAT VIDEO WEBCAM

4.1 Video Sederhana

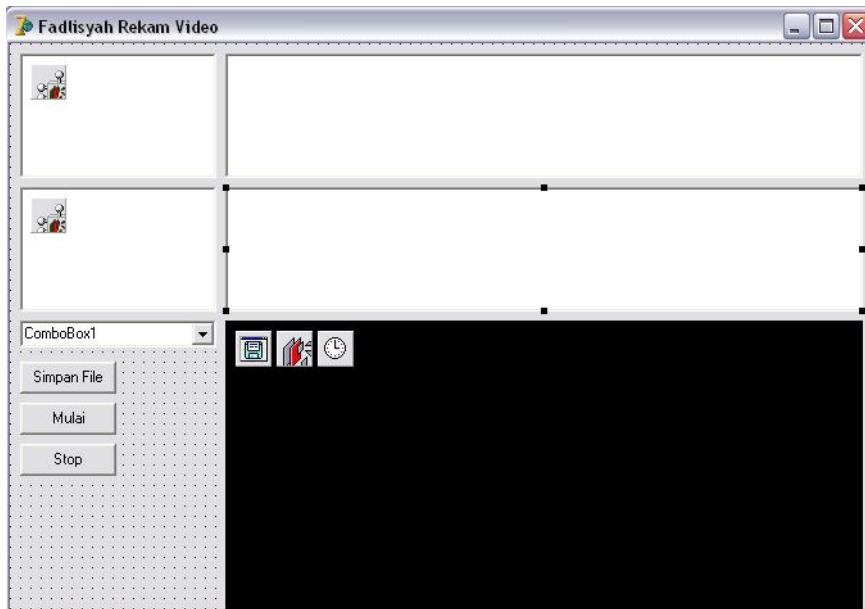
Setelah kita mempelajari penangkapan citra, selanjutnya pada bab ini saya akan menjelaskan tentang bagaimana merekam video melalui webcam. Untuk itu, ikuti langkah-langkah berikut :

1. Jalankan Delphi.
2. Tambahkan komponen ListBox1 , ListBox2 , ListBox3 , ListBox4 , ComboBox1 , FilterGraph1 , VideoWindow1 , SaveDialog1 , Filter1 , Timer1 , StatusBar1 , Button1 , Button2 , dan Button3  pada Form1.
3. Atur properti berbagai komponen di atas, sebagai :

Komponen	Properti	Nilai Properti
FilterGraph1 	Mode	gmCapture
	Name	FilterGraph1
	GraphEdit	True
	LinearVolume	True

VideoWindow1	FilterGraph	FilterGraph1
	Mode	vmNormal
Timer1	Interval	1
	Enabled	False
Button1	Caption	Simpan File
Button2	Caption	Mulai
	Enabled	False
Button3	Caption	Stop
	Enabled	False
Filter1	FilterGraph	
Filter2	FilterGraph	

4. Tampilan Form yang diharapkan akan berupa :



5. Klik 2x pada Form1, lalu tuliskan

```
procedure TForm1.FormCreate(Sender: TObject);
var i: integer;
begin
  CapEnum := 
TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  for i := 0 to CapEnum.CountFilters - 1 do
    ListBox1.Items.Add(CapEnum.Filters[i].FriendlyName);

CapEnum.SelectGUIDCategory(CLSID_AudioInputDeviceCategory);
  for i := 0 to CapEnum.CountFilters - 1 do
    ListBox3.Items.Add(CapEnum.Filters[i].FriendlyName);

VideoMediaTypes := TEnumMediaType.Create;
  AudioMediaTypes := TEnumMediaType.Create;
end;
```

6. Klik 2x pada ListBox1, lalu tuliskan kode berikut :

```
procedure TForm1.ListBox1Click(Sender: TObject);
var
  PinList: TPinList;
  i: integer;
begin
  CapEnum.SelectGUIDCategory(CLSID_VideoInputDeviceCategory);
  if ListBox1.ItemIndex <> -1 then
    begin
      Filter1.BaseFilter.Moniker :=
CapEnum.GetMoniker(ListBox1.ItemIndex);
      Filter1.FilterGraph := FilterGraph1;
      FilterGraph1.Active := true;
      PinList := TPinList.Create(Filter1 as IBaseFilter);
      ListBox2.Clear;
      VideoMediaTypes.Assign(PinList.First);
      for i := 0 to VideoMediaTypes.Count - 1 do

        ListBox2.Items.Add(VideoMediaTypes.MediaDescription[i]);
        FilterGraph1.Active := false;
        PinList.Free;
        Button1.Enabled := true;
    end;
end;
```

7. Klik 2x pada ListBox3, lalu tuliskan kode berikut :

```
procedure TForm1.ListBox3Click(Sender: TObject);
var
  PinList: TPinList;
  i, LineIndex: integer;
  ABool: LongBool;
begin
  CapEnum.SelectGUIDCategory(CLSID_AudioInputDeviceCategory);
  if ListBox3.ItemIndex <> -1 then
    begin
      Filter2.BaseFilter.Moniker :=
      CapEnum.GetMoniker(ListBox3.ItemIndex);
      Filter2.FilterGraph := FilterGraph1;
      FilterGraph1.Active := true;
      PinList := TPinList.Create(Filter2 as IBaseFilter);
      ListBox4.Clear;
      i := 0;
      while i < PinList.Count do
        if PinList.PinInfo[i].dir = PINDIR_OUTPUT then
          begin
            AudioMediaTypes.Assign(PinList.Items[i]);
            PinList.Delete(i);
          end else inc(i);

      for i := 0 to AudioMediaTypes.Count - 1 do
        begin

      ListBox4.Items.Add(AudioMediaTypes.MediaDescription[i]);
      end;

      FilterGraph1.Active := false;
      ComboBox1.Clear;
      LineIndex := -1;
      for i := 0 to PinList.Count - 1 do
        begin
          ComboBox1.Items.Add(PinList.PinInfo[i].achName);
          with (PinList.Items[i] as IAMAudioInputMixer) do
            get_Enable(ABool);
          if ABool then LineIndex := i;
        end;
      ComboBox1.ItemIndex := LineIndex;
      PinList.Free;
      Button1.Enabled := true;
    end;
  end;
```

8. Klik 2x pada Button1, lalu tuliskan kode berikut :

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  if SaveDialog1.Execute then
  begin
    CapFile := SaveDialog1.FileName;
  end;
end;
```

9. Klik 2x pada Button2, lalu tuliskan kode berikut :

```
procedure TForm1.Button2Click(Sender: TObject);
var
  multiplexer: IBaseFilter;
  Writer: IFileSinkFilter;
  PinList: TPinList;
  i: integer;
begin
  FilterGraph1.Active := true;
  if Filter2.FilterGraph <> nil then
  begin
    PinList := TPinList.Create(Filter2 as IBaseFilter);
    i := 0;
    while i < PinList.Count do
      if PinList.PinInfo[i].dir = PINDIR_OUTPUT then
      begin
        if ListBox4.ItemIndex <> -1 then
          with (PinList.Items[i] as IAMStreamConfig) do
            SetFormat(AudioMediaTypes.Items[ListBox4.ItemIndex].AMMediaType^);
        PinList.Delete(i);
      end else inc(i);
    if ComboBox1.ItemIndex <> -1 then
      with (PinList.Items[ComboBox1.ItemIndex] as
        IAMAudioInputMixer) do
        put_Enable(true);
    PinList.Free;
  end;

  if Filter1.FilterGraph <> nil then
  begin
    PinList := TPinList.Create(Filter1 as IBaseFilter);
    if ListBox2.ItemIndex <> -1 then
      with (PinList.First as IAMStreamConfig) do
        SetFormat(VideoMediaTypes.Items[ListBox2.ItemIndex].AMMediaType^);
```

```
PinList.Free;
end;
with FilterGraph1 as IcaptureGraphBuilder2 do
begin
SetOutputFileName(MEDIASUBTYPE_Avi, PWideChar(CapFile),
multiplexer, Writer);

if Filter1.BaseFilter.DataLength > 0 then
RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as
IBaseFilter, nil , VideoWindow1 as IBaseFilter);

if Filter1.FilterGraph <> nil then
RenderStream(@PIN_CATEGORY_CAPTURE, nil, Filter1 as
IBaseFilter, nil, multiplexer as IBaseFilter);

if Filter2.FilterGraph <> nil then
begin
RenderStream(nil, nil, Filter2 as IBaseFilter,
nil, multiplexer as IBaseFilter);
end;
end;
FilterGraph1.Play;
Button3.Enabled := true;
Button2.Enabled := false;
ListBox4.Enabled := false;
ListBox3.Enabled := false;
ListBox2.Enabled := false;
ListBox1.Enabled := false;
Timer1.Enabled := true;
end;
```

10. Klik 2x pada Button3, lalu tuliskan kode berikut :

```
procedure TForm1.Button3Click(Sender: TObject);
begin
Timer1.Enabled := false;
Button3.Enabled := false;
Button2.Enabled := true;
FilterGraph1.Stop;
FilterGraph1.Active := False;
ListBox4.Enabled := true;
ListBox3.Enabled := true;
ListBox2.Enabled := true;
ListBox1.Enabled := true;
end;
```

11. Klik 2x pada Timer1, lalu tuliskan kode berikut :

```
procedure TForm1.Timer1Timer(Sender: TObject);
var
  position: int64;
  Hour, Min, Sec, MSec: Word;
  const MiliSecInOneDay = 86400000;
begin
  if FilterGraph1.Active then
  begin
    with FilterGraph1 as IMediaSeeking do
      GetCurrentPosition(position);
    DecodeTime(position div 10000 / MiliSecInOneDay, Hour, Min, Sec, MSec);
    StatusBar1.SimpleText := Format('%d:%d:%d:%d',[Hour, Min, Sec, MSec]);
  end;
end;
```

12. Klik 1x pada Form1, klik tab Events dalam Object Inspector, lalu klik 2x pada sel di sebelah item OnDestroy, dan tuliskan kode berikut :

```
procedure TForm1.FormDestroy(Sender: TObject);
begin
  CapEnum.Free;
  VideoMediaTypes.Free;
  AudioMediaTypes.Free;
end;
```

13. Periksa kembali keseluruhan listing yang dibuat dengan listing berikut :

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, DSPack, DSUtil, DirectShow9, ComCtrls, ExtCtrls;

type
  TForm1 = class(TForm)
```

```
ListBox1: TListBox;
ListBox2: TListBox;
ListBox3: TListBox;
ListBox4: TListBox;
ComboBox1: TComboBox;
Button1: TButton;
Button2: TButton;
Button3: TButton;
SaveDialog1: TSaveDialog;
VideoWindow1: TVideoWindow;
FilterGraph1: TFilterGraph;
Timer1: TTimer;
Filter1: TFilter;
Filter2: TFilter;
StatusBar1: TStatusBar;
procedure FormCreate(Sender: TObject);
procedure ListBox1Click(Sender: TObject);
procedure ListBox3Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure FormDestroy(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;
  CapEnum: TSysDevEnum;
  VideoMediaTypes, AudioMediaTypes: TEnumMediaType;
  CapFile: WideString = 'c:\Fadlisyah.avi';

implementation

{$R *.dfm}
```

```
procedure TForm1.FormCreate(Sender: TObject);
var i: integer;
begin
  CapEnum := TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  for i := 0 to CapEnum.CountFilters - 1 do
    ListBox1.Items.Add(CapEnum.Filters[i].FriendlyName);

  CapEnum.SelectGUIDCategory(CLSID_AudioInputDeviceCategory);
  for i := 0 to CapEnum.CountFilters - 1 do
    ListBox3.Items.Add(CapEnum.Filters[i].FriendlyName);

  VideoMediaTypes := TEnumMediaType.Create;
  AudioMediaTypes := TEnumMediaType.Create;
end;

procedure TForm1.ListBox1Click(Sender: TObject);
var
  PinList: TPinList;
  i: integer;
begin
  CapEnum.SelectGUIDCategory(CLSID_VideoInputDeviceCategory);
  if ListBox1.ItemIndex <> -1 then
  begin
    Filter1.BaseFilter.Moniker := CapEnum.GetMoniker(ListBox1.ItemIndex);
    Filter1.FilterGraph := FilterGraph1;
    FilterGraph1.Active := true;
    PinList := TPinList.Create(Filter1 as IBaseFilter);
    ListBox2.Clear;
    VideoMediaTypes.Assign(PinList.First);
    for i := 0 to VideoMediaTypes.Count - 1 do
      ListBox2.Items.Add(VideoMediaTypes.MediaDescription[i]);
    FilterGraph1.Active := false;
    PinList.Free;
    Button2.Enabled := true;
  end;
end;

procedure TForm1.ListBox3Click(Sender: TObject);
var
```

```
PinList: TPinList;
i, LineIndex: integer;
ABool: LongBool;
begin
CapEnum.SelectGUIDCategory(CLSID_AudioInputDeviceCategory);
if ListBox3.ItemIndex <> -1 then
begin
Filter2.BaseFilter.Moniker := CapEnum.GetMoniker(ListBox3.ItemIndex);
Filter2.FilterGraph := FilterGraph1;
FilterGraph1.Active := true;
PinList := TPinList.Create(Filter2 as IBaseFilter);
ListBox4.Clear;
i := 0;
while i < PinList.Count do
if PinList.PinInfo[i].dir = PINDIR_OUTPUT then
begin
AudioMediaTypes.Assign(PinList.Items[i]);
PinList.Delete(i);
end else inc(i);

for i := 0 to AudioMediaTypes.Count - 1 do
begin
ListBox4.Items.Add(AudioMediaTypes.MediaDescription[i]);
end;

FilterGraph1.Active := false;
ComboBox1.Clear;
LineIndex := -1;
for i := 0 to PinList.Count - 1 do
begin
ComboBox1.Items.Add(PinList.PinInfo[i].achName);
with (PinList.Items[i] as IAMAudioInputMixer) do get_Enable(ABool);
if ABool then LineIndex := i;
end;
ComboBox1.ItemIndex := LineIndex;
PinList.Free;
Button2.Enabled := true;
end;
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  if SaveDialog1.Execute then
  begin
    CapFile := SaveDialog1.FileName;
    //Label1.Caption := 'c:\Fadlisyah.avi';
  end;
end;

procedure TForm1.Button2Click(Sender: TObject);
var
  multiplexer: IBaseFilter;
  Writer: IFileSinkFilter;
  PinList: TPinList;
  i: integer;
begin
  // Activate the filter graph, at this stage the source filters are added to the
  graph
  FilterGraph1.Active := true;

  // configure output Audio media type + source
  if Filter2.FilterGraph <> nil then
  begin
    PinList := TPinList.Create(Filter2 as IBaseFilter);
    i := 0;
    while i < PinList.Count do
      if PinList.PinInfo[i].dir = PINDIR_OUTPUT then
        begin
          if ListBox4.ItemIndex <> -1 then
            with (PinList.Items[i] as IAMStreamConfig) do
              SetFormat(AMMediaType^.Items[ListBox4.ItemIndex].AMMediaType);
              PinList.Delete(i);
              end else inc(i);
        if ComboBox1.ItemIndex <> -1 then
          with (PinList.Items[ComboBox1.ItemIndex] as IAMAudioInputMixer) do
            put_Enable(true);
```

```
PinList.Free;
end;

// configure output Video media type
if Filter1.FilterGraph <> nil then
begin
  PinList := TPinList.Create(Filter1 as IBaseFilter);
  if ListBox2.ItemIndex <> -1 then
    with (PinList.First as IAMStreamConfig) do

SetFormat(VideoMediaTypes.Items[ListBox2.ItemIndex].AMMediaType^);
  PinList.Free;
end;

// now render streams
with FilterGraph1 as IcaptureGraphBuilder2 do
begin
  // set the output filename
  SetOutputFileName(MEDIASUBTYPE_Avi, PWideChar(CapFile),
multiplexer, Writer);

  // Connect Video preview (VideoWindow)
  if Filter1.BaseFilter.DataLength > 0 then
    RenderStream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as IBaseFilter,
nil , VideoWindow1 as IBaseFilter);

  // Connect Video capture streams
  if Filter1.FilterGraph <> nil then
    RenderStream(@PIN_CATEGORY_CAPTURE, nil, Filter1 as IBaseFilter,
nil, multiplexer as IBaseFilter);

  // Connect Audio capture streams
  if Filter2.FilterGraph <> nil then
begin

  RenderStream(nil, nil, Filter2 as IBaseFilter,
nil, multiplexer as IBaseFilter);
end;
end;
```

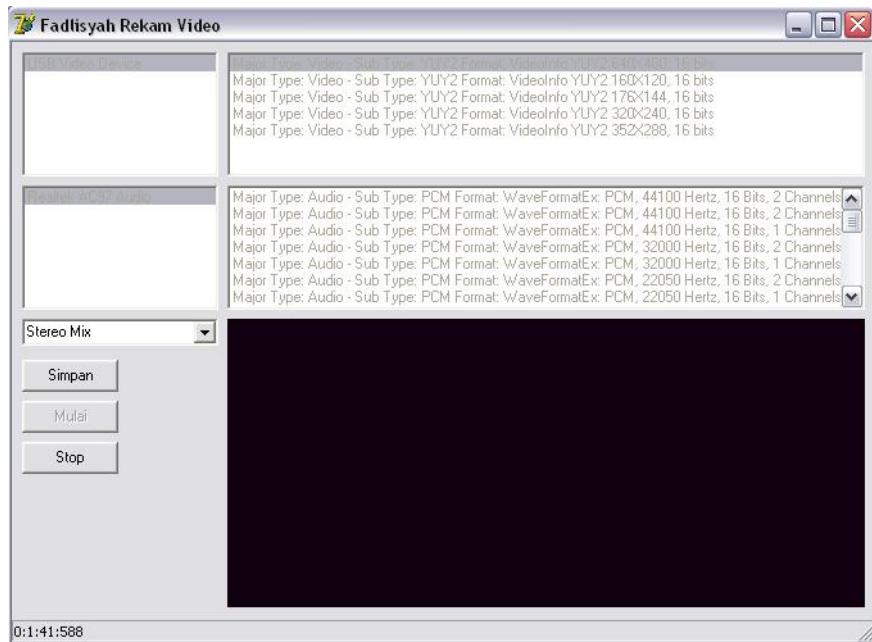
```
FilterGraph1.Play;
Button3.Enabled := true;
Button2.Enabled := false;
ListBox4.Enabled := false;
ListBox3.Enabled := false;
ListBox2.Enabled := false;
ListBox1.Enabled := false;
Timer1.Enabled := true;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
var
  position: int64;
  Hour, Min, Sec, MSec: Word;
  const MiliSecInOneDay = 86400000;
begin
  if FilterGraph1.Active then
    begin
      with FilterGraph1 as IMediaSeeking do
        GetCurrentPosition(position);
      DecodeTime(position div 10000 / MiliSecInOneDay, Hour, Min, Sec, MSec);
      StatusBar1.SimpleText := Format('%d:%d:%d.%d',[Hour, Min, Sec, MSec]);
    end;
end;

procedure TForm1.Button3Click(Sender: TObject);
begin
  Timer1.Enabled := false;
  Button3.Enabled := false;
  Button2.Enabled := true;
  FilterGraph1.Stop;
  FilterGraph1.Active := False;
  ListBox4.Enabled := true;
  ListBox3.Enabled := true;
  ListBox2.Enabled := true;
  ListBox1.Enabled := true;
end;
```

```
procedure TForm1.FormDestroy(Sender: TObject);
begin
  CapEnum.Free;
  VideoMediaTypes.Free;
  AudioMediaTypes.Free;
end;
end.
```

14. Eksekusi program dengan menekan icon . Salah satu tampilan program hasil eksekusi penulis adalah :

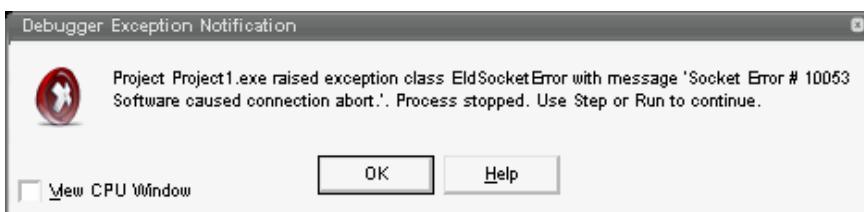


Bab 5

MEMBUAT STREAMING VIDEO di WEBSITE

5.1 Delphi Site

Setelah kita mempelajari penangkapan citra, merekam video melalui webcam, maka pada bab ini kita akan mencoba membuat sebuah streaming video di website yang sederhana. Biasanya aplikasi ini digunakan dalam system pemantauan/monitoring misalnya monitoring kepadatan lalu lintas melalui website. Untuk upload image ke server digunakan IdFTP, tapi perlu dicatat komponen ini bisa digunakan untuk upload ke server yang mempunyai OS sama dengan client, jika OS - nya berbeda akan muncul error seperti pada gambar dibawah ini



Untuk membuat video streaming dapat mengikuti langkah-langkah berikut :

1. Jalankan Delphi.

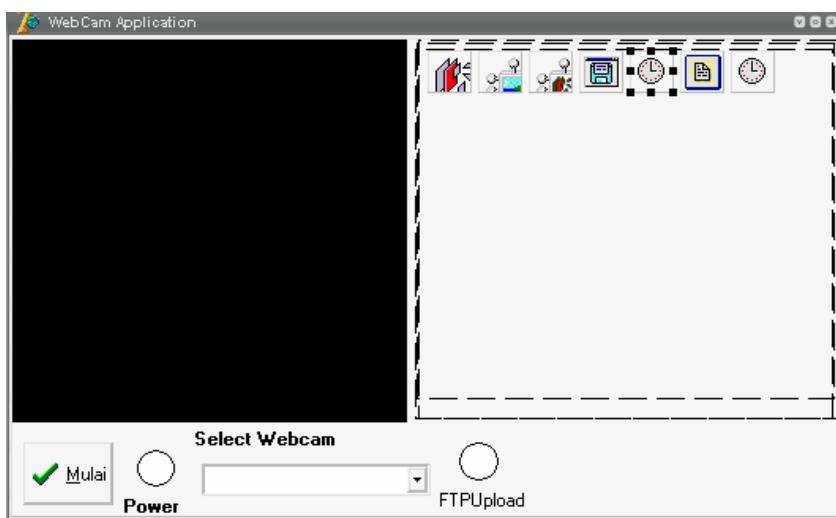
2. Tambahkan komponen Shape1 , Shape2 , Timer1 , Timer2 , IdFTP1 , FilterGraph1 , VideoWindow1 , SampleGrabber1 , SaveDialog1 , Filter1 , Image1 , Image2 , Image3 , BitBtn1 , ComboBox1 , Label1 , Label2 , Label3  pada Form1.
3. Atur properti berbagai komponen di atas, sebagai :

Komponen	Properti	Nilai Properti
	Mode	gmCapture
	Name	FilterGraph1
	GraphEdit	True
	LinearVolume	True
	FilterGraph	FilterGraph1
	Mode	vmNormal
	FilterGraph	FilterGraph1
	FilterGraph	FilterGraph1
	Host	Athena
	Password	1
	Username	Dayat
	Interval	1000
	Enabled	False
	Interval	1000
	Enabled	False
	Shape	StCircle
	Shape	StCircle

ComboBox1		Text	
BitBtn1		Caption	&Mulai
		Kind	bkOK
Label1		Caption	Power
Label2		Caption	Select WebCam
Label3		Caption	FTPUpload

Untuk IdFTP1 pengisian Host, Password dan Username disesuaikan dengan user ID saat login pada server yang digunakan.

4. Tampilan Form yang diharapkan akan berupa :



5. Klik 2x pada BitBtn1, lalu tuliskan

```
procedure TForm1.BitBtn1Click(Sender: TObject);
var
  i : integer;
```

```

begin
if bitbtn1.Caption = '&Mulai' then
begin
  shape1.Brush.Color:=clred;
  bitbtn1.Kind:=bkclose;
  combobox1.Enabled:=true;
  CapEnum:=
TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
  for i := 0 to CapEnum.CountFilters - 1 do
    begin
      combobox1.Items.Add(CapEnum.Filters[i].FriendlyName);
    end;
end
else
close;
end;

```

6. Pada Event OnClick pada ComboBox1 tuliskan procedure OnSelectDevice, lalu tuliskan kode berikut :

```

procedure TForm1.OnSelectDevice(sender: TObject);
begin
  timer2.Enabled:=true;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
  Filter1.BaseFilter.Moniker :=
CapEnum.GetMoniker(combobox1.ItemIndex);
  FilterGraph1.Active := true;
  with FilterGraph1 as ICaptureGraphBuilder2 do
    Renderstream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as
IBaseFilter, SampleGrabber1 as IBaseFilter, VideoWindow1 as
IBaseFilter);
  FilterGraph1.Play;
end;

```

7. Klik 2x pada Timer1, lalu tuliskan kode berikut :

```

procedure TForm1.Timer1Timer(Sender: TObject);
begin
  samplegrabber1.GetBitmap(Image1.Picture.Bitmap);
  image2.Picture:=image1.Picture;
  image3.picture:=image2.picture;
  timer2.Enabled:=true;
  shape2.Brush.Color:=clwhite;
// timer2.enabled:=false;
end;

```

8. Klik 2x pada Timer2, lalu tuliskan kode berikut :

```
procedure TForm1.Timer2Timer(Sender: TObject);
begin
  image3.Picture.SaveToFile('E:\test\foto.JPG');
  timer2.Enabled:=false;
  idFTP1.Connect();
  if idFTP1.Connected then
  begin
    shape2.Brush.Color:=clred;
    //TransferDirectory('E:\test\',true);
    idFTP1.Put('E:\test\foto.JPG','foto.JPG');
  end;
  idFTP1.Disconnect;
end;
```

9. Klik 1x pada Form1, klik tab Events dalam Object Inspector, lalu pada sel di sebelah item OnCloseQuery tulis procedure formclosequery, dan tuliskan kode berikut :

```
procedure TForm1.formclosequery(Sender: TObject; var
CanClose: Boolean);
begin
  capenum.Free;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
end;
```

10. Periksa kembali keseluruhan listing yang dibuat dengan listing berikut :

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, Buttons, ComCtrls, ExtCtrls, CPort, Menus, Spin, jpeg,
DSPack, DirectShow9, DSUtil, CPortCtl, Math, ExtDlgs, OleCtrls,
IdBaseComponent, IdComponent, IdTCPConnection,
IdTCPClient, IdFTP;
```

```
type
```

```
TForm1 = class(TForm)
```

```
Panel2: TPanel;
FilterGraph1: TFilterGraph;
VideoWindow1: TVideoWindow;
Image1: TImage;
SampleGrabber1: TSampleGrabber;
Filter1: TFilter;
ComboBox1: TComboBox;
Shape1: TShape;
Label1: TLabel;
Timer1: TTimer;
Image2: TImage;
Label2: TLabel;
SaveDialog1: TSaveDialog;
Timer2: TTimer;
Image3: TImage;
BitBtn1: TBitBtn;
IdFTP1: TIdFTP;
Shape2: TShape;
Label3: TLabel;
procedure OnSelectDevice(sender: TObject);
procedure formclosequery(Sender: TObject; var CanClose: Boolean);
procedure Timer1Timer(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure ComboBox1Change(Sender: TObject);
procedure FormCreate(Sender: TObject);
//procedure TransferDirectory(ADir: String; ASubDirs: Boolean = True);

private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form1: TForm1;
  CapEnum: TSysDevEnum;
implementation
  uses unit2,unit_gambar,unit_panel;
```

```
{$R *.dfm}

procedure TForm1.OnSelectDevice(sender: TObject);
begin
  timer2.Enabled:=true;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
  Filter1.BaseFilter.Moniker := CapEnum.GetMoniker(comboBox1.ItemIndex);
  FilterGraph1.Active := true;
  with FilterGraph1 as ICaptureGraphBuilder2 do
    Renderstream(@PIN_CATEGORY_PREVIEW, nil, Filter1 as
    IBaseFilter, SampleGrabber1 as IBaseFilter, VideoWindow1 as IbaseFilter);
  FilterGraph1.Play;
end;

procedure TForm1.formclosequery(Sender: TObject; var CanClose: Boolean);
begin
  capenum.Free;
  FilterGraph1.ClearGraph;
  FilterGraph1.Active := false;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
  samplegrabber1.GetBitmap(Image1.Picture.Bitmap);
  image2.Picture:=image1.Picture;
  image3.picture:=image2.picture;
  timer2.Enabled:=true;
  shape2.Brush.Color:=clwhite;
  // timer2.enabled:=false;
end;

procedure TForm1.BitBtn1Click(Sender: TObject);
var
  i : integer;
begin
  if bitbtn1.Caption = '&Mulai' then
    begin
```

```
shape1.Brush.Color:=clred;
bitbtn1.Kind:=bkclose;
comboBox1.Enabled:=true;
CapEnum:= TSysDevEnum.Create(CLSID_VideoInputDeviceCategory);
for i := 0 to CapEnum.CountFilters - 1 do
begin
  comboBox1.Items.Add(CapEnum.Filters[i].FriendlyName);
end;
else
close;
end;

//File Transfer include directory
{procedure TForm1.TransferDirectory(ADir: String; ASubDirs: Boolean =
True);
var
  Dir: String;
  iIndex : Integer;
  SearchRec : TSearchRec;
begin
  Dir := IncludeTrailingPathDelimiter(ADir);
  FillChar(SearchRec,SizeOf(SearchRec),0);
  if FindFirst(Dir + '*.*', faAnyFile, SearchRec) = 0 then
  begin
    try
      repeat
        if (SearchRec.Name <> '.') and (SearchRec.Name <> '..') then
        begin
          if ((SearchRec.Attr and faAnyFile > 0) or (SearchRec.Attr = 0)) then
          begin
            if ASubDirs then
            begin
              if SearchRec.Attr and faDirectory <> 0 then
              begin
                // This is a Directory, create it on the FTP Server
                // and recurse to transfer all files.
                IdFTP1.MakeDir(SearchRec.Name);
                IdFTP1.ChangeDir(SearchRec.Name);
              end;
            end;
          end;
        end;
      until FindNext(SearchRec) <> 0;
    finally
      if FindClose(SearchRec) <> 0 then
        raise;
    end;
  end;
end;
} 
```

```
TransferDirectory(Dir + SearchRec.Name);
end
else
  IdFTP1.Put(Dir + SearchRec.Name, SearchRec.Name);
end
else
  IdFTP1.Put(Dir + SearchRec.Name, SearchRec.Name);
end;
end;
until FindNext(SearchRec) <> 0;
finally
  FindClose(SearchRec);
end;
end;
end; }

procedure TForm1.Timer2Timer(Sender: TObject);
begin
image3.Picture.SaveToFile('E:\test\foto.JPG');
timer2.Enabled:=false;
idFTP1.Connect();
if idFTP1.Connected then
begin
shape2.Brush.Color:=clred;
//TransferDirectory('E:\test\'true);
idFTP1.Put('E:\test\foto.JPG','foto.JPG');
end;
idFTP1.Disconnect;
end;

end.
```

11. Eksekusi program dengan menekan icon .

5.2 HTML Site

Pada sisi pemrograman website file image yang ditampilkan akan di *refresh*. Dalam melakukan refresh diperlukan sintax seperti dibawah ini

```
<meta http-equiv="refresh" content="2">
```

Sedangkan untuk menampilkan gambar digunakan sintax seperti dibawah ini

```

```

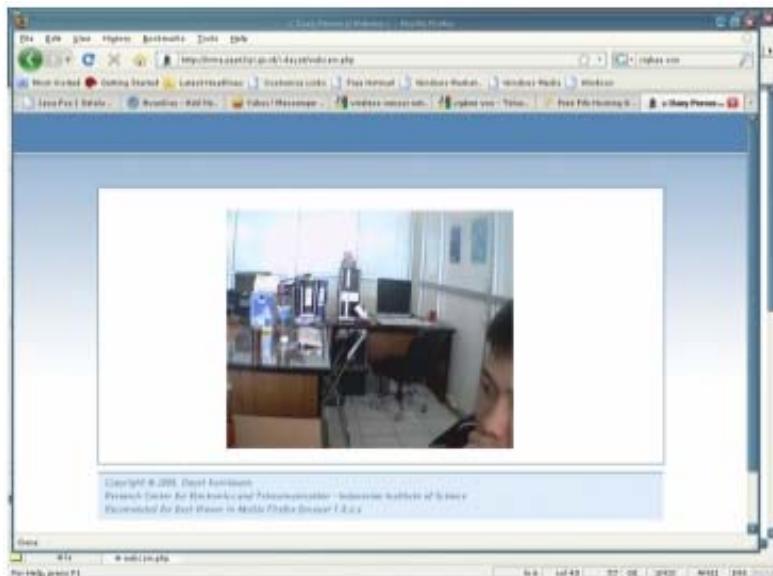
Adapun listing program lengkapnya bisa dilihat dibawah ini

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">

<head>
<title> :: Dany Personal Website :: </title>
<meta http-equiv="refresh" content="2">
<meta http-equiv="Content-Type" content="text/html;
charset=iso-8859-1">
<link rel="stylesheet" type="text/css" href="style.css"
media="screen"/>
<link rel="stylesheet" type="text/css" href="vertical.css"
media="screen"/>
<link rel="stylesheet" type="text/css" href="galery.css"
media="screen"/>
<script type="text/javascript" src="vertical.js"></script>
<script type="text/javascript" src="clock.js"></script>
<link rel="shortcut icon" href="/~dayat/images/web.ico">
</head>
<body >
<div id="content_home">
    <div id="isi_home">
        <div id="gambar_kiri">
        </div>
        <div id="gambar_tengah">
        </div>
        <div id="gambar_kanan">
        </div>
        <div id="data">
            
        </div>
        <div id="menu_home">
```

```
</div>
</div>
<div id="bawah">
<em>Copyright © 2008. Dayat Kurniawan</em><br>
<em>Research Center for Electronics and Telecommunication<br>
- Indonesian Institute of Science</em><br>
<em>Recomended for Best Viewer in Mozilla Firefox Broswer<br>
1.8.x.x</em>
</div>
</div>
</body>
</html>
<?php
?>
```

Tampilan di sisi website seperti pada gambar di bawah ini



2

DAFTAR PUSTAKA

- Fadlisyah, dkk. 2005. *Model Sistem Informasi Untuk Pengolahan Citra*, Diktat Kuliah Unimal,
- Fadlisyah. 2007. *Computer Vision & Pengolahan Citra.*, Penerbit Andi Yogyakarta, ISBN 978-979-29-0013-2.
- Fadlisyah, dkk. 2007. *Pengantar Grafika Komputer.*, Penerbit Andi Yogyakarta, ISBN 978-979-29-0060-6.
- Fadlisyah, dkk. 2008. *Pengolahan Citra Menggunakan Delphi.*, Penerbit Graha Ilmu Yogyakarta, ISBN 978-979-756-332-5.
- Fadlisyah, dkk. 2008. *Robotika : Reasoning, Learning, Planning*, Penerbit Graha Ilmu Yogyakarta.
- Fadlisyah, dkk. 2009. *Robot Visi*, Penerbit Graha Ilmu Yogyakarta.
- Fadlisyah, dkk. 2009. *Algoritma Genetika*, Penerbit Graha Ilmu Yogyakarta.
- Gonzalez, Rafael C., dan Wintz, Paul. 1987. *Digital Image Processing*, Addison Wesley
- Hearn, D. dan Baker, MP. 1994. *Computer Graphics*. Englewood Cliffs, New Jersey : Prentice-Hall
- Kadir, Abdullah. 2001. *Dasar Pemrograman Delphi 5.0*, Penerbit Andi Yogyakarta.
- Komputer, Wahana. 2003., *Panduan Praktis Pemrograman Delphi.*, Penerbit Andi Yogyakarta.
- Konishi, Scott., Yuillie, Alan L., Coughlan, James M., dan Zhu, Song Chun., 2003, *Statistical Edge Detection : Learning and Evaluating Edge Cues*, IEEE Transaction on Pattern Analysis and Machine Intelligence Vol 5, No. 1, 57 - 74
- Low, Adrian. 1991, *Computer Vision & Image Processing: Introductory*, McGraw-Hill International Editions.

Madcoms, 2003. *Pemrograman Borland Delphi 7*, Penerbit Andi Yogyakarta

Munir, Rinaldi. 2004, *Pengolahan Citra Digital dengan Pendekatan Algoritmik*, Informatika Bandung

Purcell, Edwin J. dan Varberg, Dale. 1987. *Kalkulus dan Geometri Analitis Edisi Kelima*, Erlangga

Rogers, DF dan Adams, JA.1989. *Mathematical Elements For Computer Graphic* : McGraw-Hill

<http://www.progdigy.com>

<http://awanday.wordpress.com>

RIWAYAT HIDUP PENULIS



Fadlisyah, S.Si berprofesi sebagai seorang dosen di Universitas Negeri Malikussaleh (UNIMAL). Menyelesaikan pendidikannya di Fakultas MIPA program Ilmu Komputer, Universitas Padjadjaran Bandung pada tahun 2000. Pada tahun 2001 penulis melanjutkan kuliahnya ke jenjang S2, mengambil jurusan Matematika, dan tidak menyelesaikan studinya

hingga tahun 2005 Selain aktif menulis beberapa buku teks komputer untuk tingkat bacaan mahasiswa, beliau juga aktif melakukan kegiatan riset - riset yang berkaitan dengan Artificial Intelligence, dan Computer Vision. Adapun buku - buku yang telah dikeluarkannya antara lain : Komputer Visi & Pengolahan Citra, Komputer Visi Biometriks, Robotika, Mikroprosesor, AutoCAD 2007, dan lain - lain. Mengasuh mata kuliah Komputer Grafik, Pemrograman Matematika, Teori Bahasa & Otomata, Kecerdasan Buatan, Pengolahan Citra, Komputer Visi, Mikrokontroler dan Robotika (Elektro), Pengolahan Sinyal, Komputasi Cerdas, Mikroprosesor, Metode dan Analisis Numerik, Arsitektur Komputer, dll. Pernah menerima hibah penelitian mengenai Face Detection Menggunakan Jaringan Syaraf Tiruan, Dibiayai oleh DIPA Universitas Malikussaleh, sesuai dengan surat Perjanjian Pelaksanaan Kegiatan penelitian Nomor: 20/H45/LL/2008 Tanggal 21 Mei 2008. Alamat yang dapat dihubungi : Fakultas Teknik UNIMAL, jln. Samudera No.35. Lhokseumawe. Telepon 0645- 42076.



FAHMI S.T, M.Sc. menyelesaikan program masternya di universitas **Fachhochschule Karlsruhe, Germany**, tahun 2005. Alumni Institut Teknologi Bandung ini berprofesi sebagai staf pengajar pada program S2 Sistem komputer Universitas Sumatera Utara dan juga menjabat sebagai sekretaris prodi pada konsentrasi yang sama. Pada tahun 2002 beliau telah terdaftar

sebagai anggota IEEE, dan telah memenangi beberapa award seperti **DAAD-Siemens Scholarship** Program, ASIA 21st Century contact: Heike Gabler (gabler@daad.de), **Epson Scholarship** Program, dan lain-lain. Beberapa pelatihan yang beliau ikuti adalah **ICTS 2006**, **ICTS 2007**, APCCAS2002 Asia Pacific Conference on Circuit and Systems - IEEE (*Passive participant*), MCMT2002 APT Telemedicine Workshop, dan lain-lain. Untuk konsultasi dapat menghubungi beliau melalui email fahmimin@usu.ac.id atau melalui no. +62618223371.



Dayat Kurniawan, Menyelesaikan S1 di Universitas Brawijaya Malang tahun 2002, sempat bekerja di LG-ITIN sebagai Staff RnD tuner TV, interest di bidang programming seperti java, c, Delphi, embedded programming, mikrokontroller, sekarang bekerja di **Lembaga Ilmu Pengetahuan Indonesia** di Depertemen Elektronika dan Telekomunikasi. Pernah mengikuti **Kontes Robot** Indonesia 2004-2006 dan peraih Bogasari Nugraha 2005. Beberapa artikel menarik ada di www.awanday.wordpress.com. Bisa menghubungi penulis di d_yat02@yahoo.com Hp. 081220440463.

PEMROGRAMAN DELPHI MENGGUNAKAN WEBCAM

SAAT INI PARA PEMAKAI KOMPUTER HAMPIR SELALU TERLIBAT DENGAN WEB CAMERA. WEB CAMERA YANG SERING KITA TEMUKAN SERING SEAKAN-AKAN TIDAK BERDAYA DIBANDINGKAN DENGAN KAMERA DIGITAL YANG HARGANYA RELATIF LEBIH MAHAL. KETIDAKBERDAYAAN WEB CAMERA SEBENARNYA HANYA PERSEPSI DARI PIKIRAN AWAM KITA SAJA, DAN BERSAMA BUKU INI, PENULIS AKAN MEMAPARKAN TEKNIK-TEKNIK PEMROGRAMAN UNTUK PEMBERDAYAAN WEB CAMERA.

BUKU INI HADIR UNTUK MENGGUGAH PERSEPSI KITA SELAMA INI, DENGAN MEMBERDAYAKAN KEMAMPUAN WEB CAMERA HINGGA MENCAPAI KEMAMPUAN KAMERA DIGITAL. UNTUK ITU MATERI YANG DISAJIKAN MELIPUTI : INSTALASI DSPACK, CAPTURE CITRA, PEMROGRAMAN HISTOGRAM, TEKNIK PEREKAMAN VIDEO MELALUI WEBCAM, DAN STREAMING VIDEO via WEBSITE.