

# Pengantar Steganografi

Andino Masaleno

## ***Lisensi Dokumen:***

*Copyright © 2003 - 2006 IlmuKomputer.Com*

*Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.*

## KATA PENGANTAR

Jaringan komputer dan Internet telah mengalami perkembangan yang sangat pesat. Teknologi ini mampu menyambungkan hampir semua komputer yang ada di dunia sehingga bisa saling berkomunikasi dan bertukar informasi. Bentuk informasi yang dapat ditukar berupa data teks, gambar, gambar bergerak dan suara.

Seiring dengan perkembangan tersebut, secara langsung ikut mempengaruhi cara kita berkomunikasi. Kalau dahulu untuk berkomunikasi pesan atau surat dengan menggunakan pos, sekarang telah banyak layanan *e-mail* di Internet yang dapat mengirimkan pesan secara langsung ke penerimanya. Akan tetapi sebagai suatu jaringan publik, Internet rawan terhadap pencurian data.

Steganografi sebagai suatu seni menyembunyikan pesan ke dalam pesan lainnya yang telah ada sejak sebelum masehi dan kini seiring dengan kemajuan teknologi jaringan serta perkembangan dari teknologi digital, steganografi banyak dimanfaatkan untuk mengirim pesan melalui jaringan Internet tanpa diketahui orang lain dengan menggunakan media digital berupa file gambar.

Penggunaan steganografi menjadi daya tarik banyak orang pada peristiwa penyerangan gedung WTC, 11 September 2001. Pada peristiwa tersebut disebutkan oleh "pejabat pemerintah dan para ahli dari pemerintahan AS" yang tidak disebut namanya bahwa "para teroris menyembunyikan peta-peta dan foto-foto target dan juga perintah untuk aktivitas teroris di ruang *chat sport*, *bulletin boards* porno dan *web site* lainnya". Isu lainnya menyebutkan bahwa teroris menyembunyikan pesan-pesannya dalam gambar-gambar porno di *web site* tertentu.

Buku ini membahas mengenai pengenalan steganografi serta pembuatannya dengan Delphi. Delphi dipilih dengan alasan karena menggunakan bahasa object Pascal yang telah memasyarakat bagi masyarakat awam, pelajar dan mahasiswa, terlebih mahasiswa informatika dan ilmu komputer di Indonesia.

## 1.1 Sejarah Steganografi

Steganografi merupakan seni menyembunyikan pesan ke dalam pesan lainnya sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu di dalam pesan tersebut. Kata steganografi (*steganography*) berasal dari bahasa Yunani yaitu *steganos* yang artinya tersembunyi atau terselubung dan *graphein*, yang artinya menulis, sehingga kurang lebih artinya adalah "menulis tulisan yang tersembunyi atau terselubung" (Sellars, 1996). Teknik ini meliputi banyak sekali metoda komunikasi untuk menyembunyikan pesan rahasia. Metoda ini termasuk tinta yang tidak tampak, *microdots*, pengaturan kata, tanda tangan digital, jalur tersembunyi dan komunikasi spektrum lebar.

Catatan pertama tentang steganografi ditulis oleh seorang sejarawan Yunani, Herodotus, yaitu ketika Histaeus seorang raja kejam Yunani dipenjarakan oleh Raja Darius di Susa pada abad 5

Sebelum Masehi. Histaeus harus mengirim pesan rahasia kepada anak laki-lakinya, Aristagoras, di Militus. Histaeus menulis pesan dengan cara mentato pesan pada kulit kepala seorang budak dan ketika rambut budak itu mulai tumbuh, Histaeus mengutus budak itu ke Militus untuk mengirim pesan di kulit kepalanya tersebut kepada Aristagoras.

Cerita lain tentang steganografi datang juga dari sejarawan Yunani, Herodotus, yaitu dengan cara menulis pesan pada papan kayu yang ditutup dengan lilin. Demeratus, seorang Yunani yang akan mengabarkan berita kepada Sparta bahwa Xerxes bermaksud menyerbu Yunani. Agar tidak diketahui pihak Xerxes, Demaratus menulis pesan dengan cara mengisi tabung kayu dengan lilin dan menulis pesan dengan cara mengukirnya pada bagian bawah kayu, lalu papan kayu tersebut dimasukkan ke dalam tabung kayu, kemudian tabung kayu ditutup kembali dengan lilin.

Teknik steganografi yang lain adalah tinta yang tak terlihat. Teknik ini pertama digunakan pada zaman Romawi kuno yaitu dengan menggunakan air sari buah jeruk, urine atau susu sebagai tinta untuk menulis pesan. Cara membacanya adalah dengan dipanaskan di atas nyala lilin, tinta yang sebelumnya tidak terlihat, ketika terkena panas akan berangsur-angsur menjadi gelap, sehingga pesan dapat dibaca. Teknik ini pernah juga digunakan pada Perang Dunia II.

Pada abad 20, steganografi benar-benar mengalami perkembangan. Selama berlangsung perang Boer, Lord Boden Powell (pendiri gerakan kepanduan) yang bertugas untuk membuat tanda posisi sasaran dari basis artileri tentara Boer, untuk alasan keamanan, Boden Powell menggambar peta-peta posisi musuh pada sayap kupu-kupu agar gambar-gambar peta sasaran tersebut terkamuflase.

Perang Dunia II adalah periode pengembangan teknik-teknik baru steganografi. Pada awal Perang Dunia II walaupun masih digunakan teknik tinta yang tak terlihat, namun teknik-teknik baru mulai dikembangkan seperti menulis pesan rahasia ke dalam kalimat lain yang tidak berhubungan langsung dengan isi pesan rahasia tersebut, kemudian teknik menulis pesan rahasia ke dalam pita koreksi karbon mesin ketik, dan juga teknik menggunakan pin berlubang untuk menandai kalimat terpilih yang digunakan dalam pesan, teknik terakhir adalah microdots yang dikembangkan oleh tentara Jerman pada akhir Perang Dunia II.

Dari contoh-contoh steganografi konvensional tersebut dapat dilihat bahwa semua teknik steganografi konvensional berusaha merahasiakan komunikasi dengan cara menyembunyikan pesan ataupun mengamuflase pesan. Maka sesungguhnya prinsip dasar dalam steganografi lebih dikonsentrasikan pada kerahasiaan komunikasinya bukan pada datanya (Johnson, 1995).

Seiring dengan perkembangan teknologi terutama teknologi komputasi, steganografi merambah juga ke media digital, walaupun steganografi dapat dikatakan mempunyai hubungan erat dengan kriptografi, tetapi kedua metode ini sangat berbeda.

## 1.2 Steganografi Pada Media Digital File Gambar

Pada komputer, gambar yang tampil di layar monitor merupakan kumpulan array yang merepresentasikan intensitas cahaya yang bervariasi pada pixel. Pixel adalah titik di layar monitor yang dapat diatur untuk menampilkan warna tertentu. Pixel disusun di layar monitor dalam susunan baris dan kolom. Susunan pixel dalam baris dan kolom ini yang dinamakan resolusi monitor. Resolusi monitor yang sering dijumpai adalah 640x480, 800x600, 1024x768. Resolusi 640x480 akan menampilkan pixel sejumlah 640 baris dan 480 kolom, sehingga total pixel yang digunakan  $640 \times 480 = 307.200$  pixel. Melalui pixel inilah suatu gambar dapat dimanipulasi untuk menyimpan informasi yang akan digunakan sebagai salah satu pengimplementasian steganografi.

Steganografi pada media digital file gambar digunakan untuk mengeksploitasi keterbatasan kekuatan sistem penglihatan manusia dengan cara menurunkan kualitas warna pada file gambar yang belum disisipi pesan rahasia. Sehingga dengan keterbatasan tersebut manusia sulit menemukan gradasi penurunan kualitas warna pada file gambar yang telah disisipi pesan rahasia.

# Pengenalan Borland Delphi

Borland Delphi merupakan program aplikasi database yang berbasis object Pascal dari Borland. Selain itu, Borland Delphi juga memberikan fasilitas pembuatan aplikasi visual.

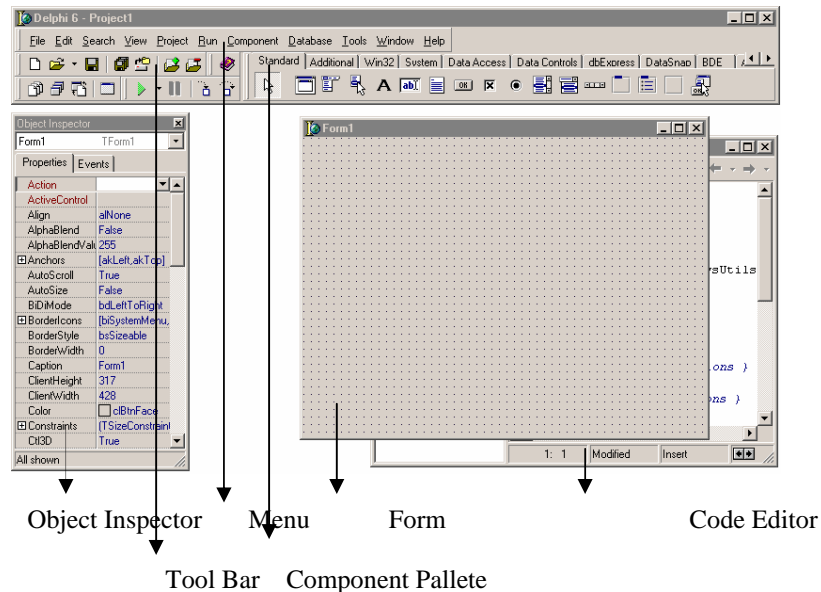
Borland Delphi memiliki komponen-komponen visual maupun non visual berintegrasi yang akan menghemat penulisan program. Terutama dalam hal perancangan antarmuka grafis (*Graphical User Interface*), kemampuan Borland Delphi untuk menggunakan Windows API (*Application Programming Interface*) ke dalam komponen-komponen visual menyebabkan pemrograman Borland Delphi yang bekerja dalam lingkungan Windows menjadi lebih mudah. Kelebihan Borland Delphi dalam hal kompilasi program juga menjadi faktor yang mempengaruhi pemilihan bahasa pemrograman yang digunakan. Karena program dikembangkan berdasarkan bahasa Pascal yang telah dikenal luas, maka untuk pengembangan program akan lebih mudah. Borland Delphi juga mempunyai kemampuan bekerja untuk pengolahan gambar dengan tersedianya unit GRAPHICS.

Warna pixel pada Borland Delphi dijabarkan dalam 4 byte hexadecimal, dengan 3 byte terendahnya adalah nilai intensitas RGB, dengan susunan sebagai berikut: \$00BBGGRR, sebagai contoh nilai dari \$00FF0000 berarti intensitas biru murni, \$0000FF00 adalah hijau murni dan \$000000FF adalah merah murni. \$00000000 adalah hitam dan \$00FFFFFF adalah putih. Untuk mengambil nilai intensitas RGB suatu pixel dapat menggunakan perintah GetRvalue, GetGvalue, GetBvalue. Contoh GetRvalue (\$000000FF) berarti mengambil nilai intensitas merah, sedang untuk menggabungkan ketiga nilai intensitas RGB tersebut dapat menggunakan perintah RGB (R, G, B) contoh RGB (\$FF, \$FF, \$FF).

Pada pembuatan aplikasi steganografi ini kita akan menggunakan Borland Delphi 6. Dalam bab ini akan dijelaskan mengenai Integrated Development Environment, rutin dan struktur kontrol pada Delphi.

## 2.1 Integrated Development Environment

Tampilan bidang kerja yang disebut dengan IDE (*Integrated Development Environment*) Delphi 6 bisa dilihat pada gambar 2.1 di bawah ini. IDE ini secara garis besar terdiri atas tiga bagian utama, yaitu Window Utama, Object Inspector dan Editor.

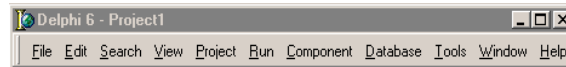


**Gambar 2.1** Tampilan IDE Delphi 6

Window utama terdiri atas Menu Bar, Tool Bar dan Component Pallette. Object Inspector menyediakan dua kelompok pengaturan komponen, yaitu Properties dan Event. Editor yang disediakan ada dua buah yaitu Form Editor dan Code Editor.

### 2.1.1 Menu Bar

Menu Bar menyediakan kelompok perintah yang digolongkan dalam sebelas menu. Menu tersebut antara lain File, Edit, Search, View, Project, Run, Component, Database, Tools, Options, dan Help.



**Gambar 2.2** Menu bar

**Tabel 2.1** Komponen Menu Bar

Nama	Keterangan
File	Kelompok perintah yang berfungsi untuk pengaturan suatu file. Misalnya New, Save, Open, dan sebagainya.
Edit	Menyimpan perintah-perintah untuk pengeditan. Mulai dari pengeditan objek, pengeditan komponen maupun pengeditan kode pada Code Editor. Contohnya Cut, Paste, Align.
Search	Kelompok perintah yang berfungsi untuk melakukan proses pencarian, baik objek, kata, kalimat dan sebagainya.
View	Menampung perintah-perintah untuk mengaktifkan bagian pendukung Integrated Development Environment.
Project	Kelompok perintah yang berfungsi untuk manajemen proyek berikut bagian-bagian pendukungnya.
Run	Kelompok perintah untuk menangani proses kompilasi program seperti Run, Build, Step Over, Debug dan seterusnya.
Component	Mengatur suatu komponen.
Database	Mengandung perintah-perintah untuk pengaturan aplikasi database.
Tools	Kelompok perintah yang berfungsi sebagai penyedia perlengkapan tambahan yang diperlukan dalam penyusunan program seperti Image Editor, Database Engine dan lain-lain.
Options	Mengatur komposisi bidang kerja Delphi.
Help	Memberikan informasi yang bersifat menolong pemakai dalam menggunakan Delphi.

### 2.1.2 Tool Bar

Tool Bar adalah bagian dari Delphi yang menyediakan tombol-tombol speed. Tombol speed fungsinya sama dengan perintah yang ada pada menu. Tombol ini disediakan dengan tujuan meringkas atau mempercepat pekerjaan kita.

Untuk mengakses suatu perintah dengan menggunakan menu, langkah yang kita lakukan adalah klik menu lalu klik submenu lantas klik perintah tersebut. Sedangkan untuk mengakses perintah dengan menggunakan tombol speed, langkah yang kita lakukan hanya mengklik tombol yang kita inginkan.

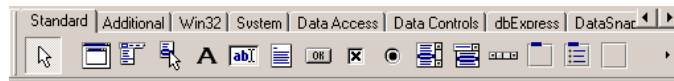


**Gambar 2.3** Tool bar

### 2.1.3 Component Pallette

Component pallette menyediakan berbagai komponen yang bisa kita pasangkan pada form sesuai keperluan kita. Telah tersedia berbagai komponen yang dikelompokkan dalam sebelas

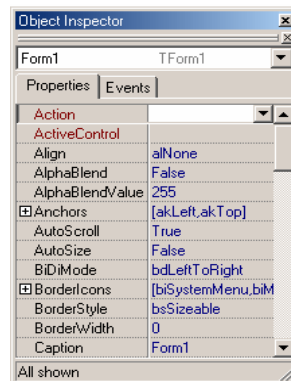
kelompok seperti tampak pada gambar.



**Gambar 2.4** Component palette

### 2.1.4 Object Inspector

Object inspector adalah sarana pengaturan objek yang kita pasang pada form, atau form itu sendiri. Dua hal penting yang bisa kita setel pada komponen adalah Properti dan event. Properti adalah yang terkait dengan sifat komponen seperti ukuran, warna dan sebagainya. Sedangkan event adalah kejadian atau peristiwa yang kita inginkan terpasang pada komponen tersebut kaitannya dengan proses pemakaian. Contoh event misalnya klik, klik ganda, drag (geser), drop dan sebagainya.

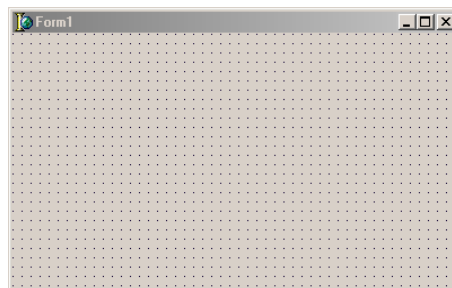


**Gambar 2.5** Object Inspector

### 2.1.5 Form

Form adalah bahan dasar yang akan menjadi jendela aplikasi kita. Pada form terdapat tiga tombol kontrol, yaitu Minimize, Maximize/Restore dan Close.

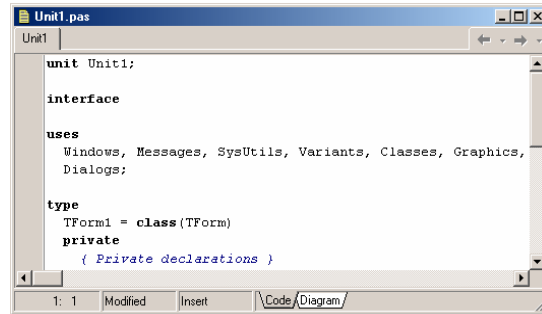
Terdapat juga caption bar tempat kita menempatkan judul Form (yang kelak menjadi judul window) dan icon. Pembatas form juga bisa diubah ukurannya dengan cara drag (geser)-drop. Pada form kita bisa meletakkan komponen-komponen yang kita perlukan dalam suatu User Interface.



**Gambar 2.6** Form

### 2.1.6 Code Editor

Code Editor adalah tempat kita menuliskan program dalam bahasa Object Pascal. Secara default Code Editor ini terletak di belakang Form Editor.



**Gambar 2.7** Code Editor

Untuk menuliskan kode yang kita pasangkan pada suatu komponen, klik ganda komponen tersebut, Code Editor akan otomatis aktif dan menempatkan kursor di lokasi penulisan kode.

## 2.2 Membuat Program Sederhana

Sampai di sini kita telah mempelajari perlengkapan yang disediakan dalam Integrated Development Environment Delphi, sekarang marilah kita mencoba menyusun program sederhana dengan perlengkapan tersebut.

### 2.2.1 Menjalankan Delphi

Sebelum membangun suatu aplikasi, siapkanlah Delphi. Instalasikan ke hard disk Anda jika memang belum ada dalam hard disk atau untuk kelengkapan. Kemudian jalankan Delphi, ada beberapa cara yang bisa dipilih, yaitu:

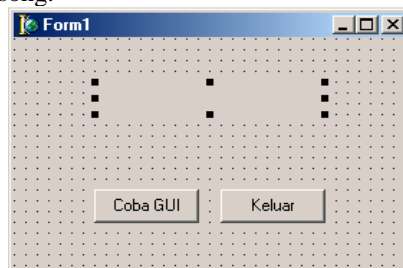
- ☐ Klik ganda icon Delphi 6
- ☐ Jalankan Windows Explorer atau File Manager untuk menampilkan file-file yang ada pada hard disk komputer Anda. Kemudian klik ganda file bernama delphi32.EXE. (Jika pada saat instalasi Anda tidak mengubah susunan folder maka file tersebut akan terletak pada folder dengan path \Program Files\Borland\Delphi6\Bin).
- ☐ Dari tombol Start Windows pilihlah Run lalu pilihlah file delphi32.exe sesuai path di atas lalu klik OK.
- ☐ Pengaktifan salah satu cara di atas akan menjalankan Delphi 6 yang tampilan awalnya tampak seperti gambar sebelumnya.

### 2.2.2 Membuat Aplikasi GUI

Yang dimaksud Aplikasi GUI adalah aplikasi yang beroperasi pada sistem operasi Windows 95 atau Windows NT. Misalnya aplikasi ini kita harapkan bisa menampilkan sebuah teks apabila sebuah tombol ditekan. Langkah-langkah penyusunannya adalah:

#### Merancang Form

Form yang kita harapkan misalnya seperti Gambar 1.16 yang terdiri atas tombol bernama TombolCoba berjudul Coba GUI, tombol bernama Tombol Keluar berjudul Keluar, label bernama Label1 dan judul (Caption) kosong.



**Gambar 2.8** Rancangan Form

Rincian masing-masing komponen yang ada pada rancangan form di atas dapat dilihat pada tabel 2.2 di bawah ini.

**Tabel 2.2** Rincian komponen form Menu Utama

Komponen	Properti	Nilai
Form1	Caption	Form1
	Name	Form1
Label1	Caption	
	Name	Label1
Button1	Caption	Coba GUI
	Name	BtnCoba
Button2	Caption	Keluar
	Name	Btn Keluar

## Memasang Code Program

Setelah memasang komponen dan mengatur propertinya, pasangkanlah kode program yang diperlukan dengan langkah-langkah:

1. Klik ganda TombolCoba, Code Editor akan aktif dan kursor berada pada titik penulisan kode.

2. Lalu isikan kode berikut:

```
Label1.Caption:= 'Coba aplikasi GUI';
```

3. Klik ganda TombolKeluar lalu isikan kode berikut ini:

```
Close;
```

4. Sehingga kode keseluruhannya adalah:

```
unit Unit1;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls;
type
    TForm1 = class(TForm)
        TombolCoba: TButton;
        TombolKeluar: TButton;
        Label1: TLabel;
        procedure TombolCobaClick(Sender: TObject);
        procedure TombolKeluarClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.TombolCobaClick(Sender: TObject);
begin
    Label1.Caption:= 'Coba aplikasi GUI';
end;

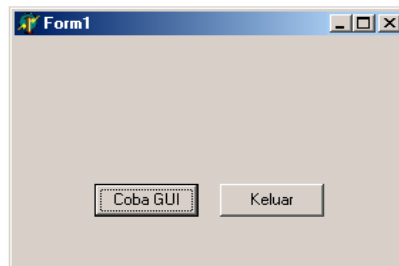
procedure TForm1.TombolKeluarClick(Sender: TObject);
begin
    Close;
end;

end.
```

## Menjalankan Program

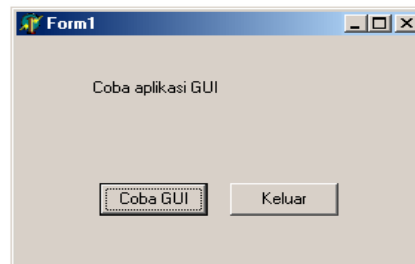
Untuk menjalankan program caranya sangat sederhana, yaitu:

1. Klik tombol Run pada Tool Bar, program akan dijalankan seperti gambar 2.9 berikut ini.



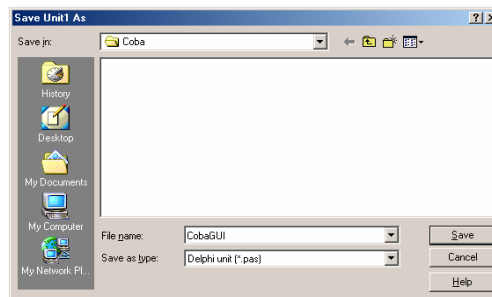
**Gambar 2.9** Program dijalankan

- Setelah program dijalankan lalu klik tombol Coba GUI, hasilnya akan tampak seperti gambar 2.10 berikut ini.

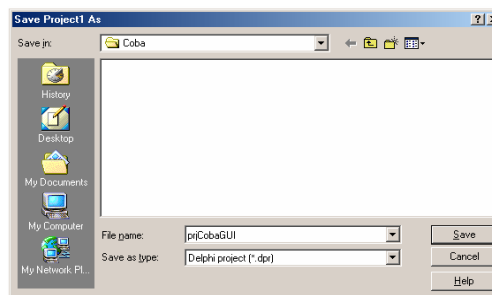


**Gambar 2.10** Tampilan program

- Klik tombol Keluar, program akan tertutup.
- Anda bisa menyimpan program dengan nama tertentu, dengan perintah Save Project As.. dari menu File. Delphi 6 akan menampilkan kotak dialog Save Project As. Juga simpanlah file unit dengan perintah File|Save As yang akan menampilkan kotak dialog Save Unit As.



**Gambar 2.11** Kotak dialog Save Unit As



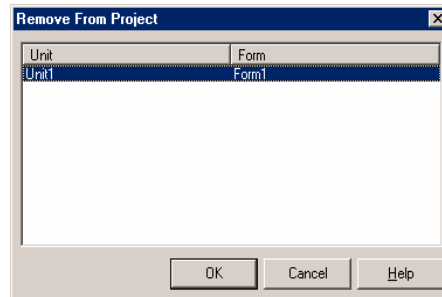
**Gambar 2.12** Kotak dialog Save Project As

5. Kemudian isikanlah nama proyek maupun nama file unitnya pada File Name dan lokasi driver/folder penyimpanannya pada Save In. Lalu klik tombol Save.

### 2.2.3 Membuat Aplikasi Console

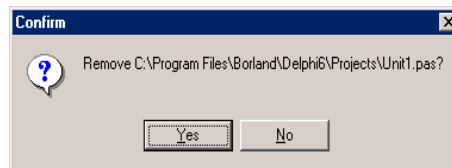
Aplikasi Console adalah aplikasi dengan mode teks seperti aplikasi DOS, bukan mode grafik seperti Windows/GUI. Delphi 6 tidak menyediakan lagi sarana akses pembangunan aplikasi console. Langkah pembuatan aplikasi console adalah:

1. Aktifkan perintah File | New Application
2. Klik tombol Remove file from Project pada Tool Bar. Kemudian muncul kotak dialog seperti gambar 2.13 berikut.



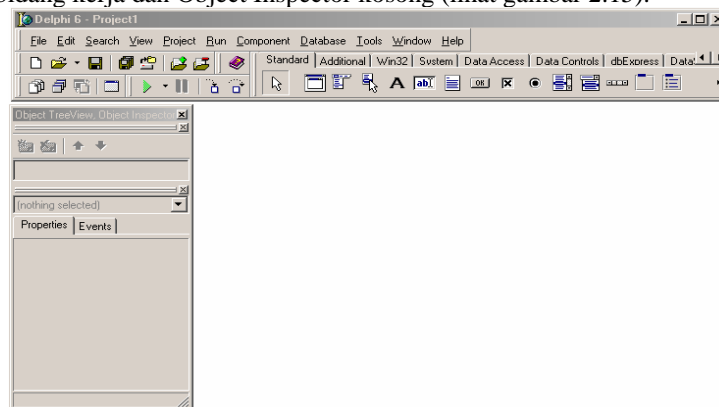
**Gambar 2.13** Kotak dialog Remove From Project

3. Klik OK. Muncul konfirmasi remove unit seperti gambar 2.14.



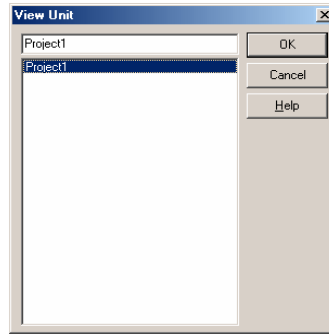
**Gambar 2.14** kotak dialog remove unit

4. Klik Yes, bidang kerja dan Object Inspector kosong (lihat gambar 2.15).



**Gambar 2.15** Bidang kerja dan object inspector kosong

5. Sekarang kita tampilkan kode project. Aktifkan perintah View | Units. Tampil kotak dialog View Units (lihat Gambar 2.16).



**Gambar 2.16** Kotak dialog View Units

6. Klik Project 1, lalu klik OK. Muncullah Code Editor yang menampilkan kode program berikut:

```
program Project1;
uses
  Forms;                {Gantilah baris ini}
{$R *.res}
{$APPTYPECONSOLE} {Tambahkan Baris ini}
begin
  Application.Initialize;
  Application.Run;
end.
```

7. Kode di atas itulah yang akan kita gunakan untuk menyusun program console. Selanjutnya kita lakukan perubahan beberapa baris kode seperti tertera pada tabel 2.3 berikut ini.

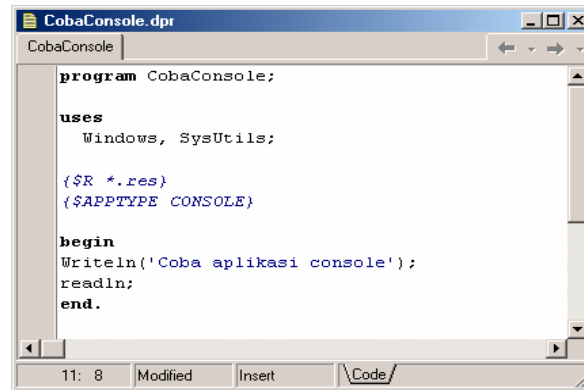
**Tabel 2.3** Daftar Koreksi

Kode	Pembetulan
Forms;	Gantilah dengan Windows, SysUtils;
Setelah {\$R*.RES}	Tambahkan {\$APPTYPE CONSOLE}
Application.Initialize;	Gantilah sesuai keperluan.
Application.Run;	Gantilah sesuai keperluan.

8. Selanjutnya kita bisa mengganti kedua baris kode yang ada pada blok begin..end; dengan kode sesuai dengan keperluan kita. Misalnya seperti berikut ini:

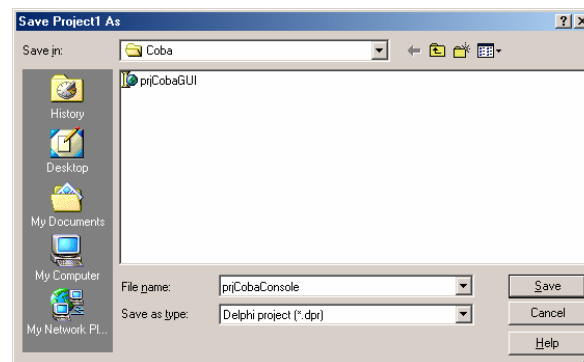
```
begin
  Writeln ('Coba aplikasi console');
  Readln;
end;
```

9. Kode program selengkapnya adalah seperti gambar 2.17 berikut ini.



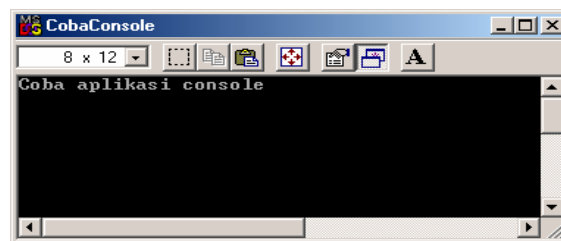
**Gambar 2.17** Contoh Aplikasi Console

10. Kemudian simpanlah file program ini dengan nama tertentu. Caranya, aktifkan perintah File | Save As..atau Save Project As...Muncul kotak dialog Save Project As...seperti Gambar 2.18.



**Gambar 2.18** Kotak dialog Save Project As

11. Pada kotak dialog tersebut isikan nama file. Lalu klik Save.
12. Kemudian jalankan program, klik tombol Run pada Tool Bar. Program akan dijalankan seperti gambar 2.19.



**Gambar 2.19** Program CobaConsole.exe dijalankan

## 2.3 Rutin Standar

Delphi memiliki fasilitas RTL (*Run Time Library*). RTL adalah koleksi atau pustaka yang menyimpan bermacam-macam rutin. Koleksi ini memang tidak tampak secara visual namun banyak manfaatnya.

Rutin yang tersimpan pada RTL menyediakan sarana untuk berbagi keperluan, seperti pengaturan waktu, pengaturan file, pengaturan objek eksepsi, penghitungan matematika sampai dengan pengaturan memori dan lain-lain. Pada tabel 2.4 di bawah, disajikan sebagian rutin standar pada Delphi.

**Tabel 2.4** Sebagian rutin standar pada Delphi

<b>Rutin</b>	<b>Keterangan</b>
Chr	Menghasilkan karakter dengan angka ordinal khusus.
Copy	Menghasilkan substring suatu string.
Date	Menghasilkan tanggal saat itu.
DateTimeToStr	Mengubah nilai tanggal dan waktu menjadi sebuah string dengan format default.
DateToStr	Mengubah nilai tanggal menjadi string.
Dec	Menurunkan nilai variable yang digunakan sebagai parameter. Penurunan sebesar satu atau tergantung dari nilai parameter kedua.
Delete	Menghapus sebuah substring dari sebuah string.
FloatToStr	Mengubah nilai floating point menjadi string padanannya dalam format default.
FloatToStrF	Mengubah nilai floating point menjadi string padanannya dalam format khusus.
FormatDateTime	Memformat nilai tanggal dan waktu dengan format khusus.
High	Menentukan nilai tertinggi di dalam range dari data tipe ordinal yang diberikan.
Inc	Menaikkan nilai variabel yang digunakan sebagai parameter. Penaikan sebesar satu atau tergantung dari nilai parameter yang telah ditentukan.
Int	Menghasilkan bagian-bagian Integer dari nilai argumen floating point.
IntToStr	Mengubah sebuah angka (integer) menjadi sebuah string
Length	Menghasilkan panjang string yang dinamis.
Low	Menentukan nilai terendah di dalam range dari data tipe ordinal yang diberikan.
Now	Menghasilkan waktu dan tanggal yang sedang berlangsung pada saat itu.
Ord	Menghasilkan sebuah angka yang menunjukkan elemen atau argumen dalam himpunan tipe data/menghasilkan angka ordinal.
Pred	Menghasilkan predecessor (proses pengawal) suatu nilai. Kita bisa menggunakan fungsi ini sebagai penghasil run-time error yang bisa kita tangani dengan menggunakan exception.
Round	Mengubah sebuah nilai tipe real menjadi integer, dengan membulatkan sisanya.
Str	Mengubah nilai numerik menjadi string.
StrToDate	Mengubah string menjadi nilai tanggal.
StrToDateTime	Mengubah string menjadi tanggal dan waktu.
StrToFloat	Mengubah nilai string Pascal menjadi sebuah nilai floating point.
StrToInt	Mengubah sebuah string menjadi sebuah angka dengan menampilkan eksepsi jika stringnya salah.
StrToTime	Mengubah string menjadi nilai waktu.
Time	Menghasilkan waktu saat itu.
Succ	Menghasilkan successor (proses penerus) suatu nilai. Kita bisa menggunakan fungsi ini sebagai penghasil run-time error yang bisa kita tangani dengan menggunakan exception
TimeToStr	Mengubah nilai waktu menjadi string.
Trunc	Mengubah nilai tipe Real menjadi integer, dengan memotong sisanya.
UpperCase	Mengubah semua karakter string menjadi huruf besar.

## 2.4 Struktur Kontrol

Setiap bahasa pemrograman pasti memiliki struktur kontrol, demikian juga dengan Delphi. Struktur kontrol berupa operasi kondisional untuk pemilihan suatu aksi dan operasi pengulangan untuk melakukan perulangan pada suatu aksi. Contoh-contoh program pada struktur kontrol ini akan dijalankan pada aplikasi console.

## 2.4.1 Operasi Kondisional

### 2.4.1.1 If-Then-Else

Statement If-Then-Else digunakan untuk mengeksekusi sebuah blok jika memenuhi kondisi tertentu. Bila kondisi yang diseleksi terpenuhi, maka blok yang mengikuti Then akan diproses, sebaliknya bila kondisi tidak terpenuhi, maka yang akan diproses blok berikutnya. Sintaks penggunaannya adalah :

```
If Kondisi Then [blok perintah1]  
Else [blok perintah2]
```

```
program CobalIfThenElse;  
uses  
    Windows, SysUtils;  
  
{ $R *.res }  
{ $APPTYPE CONSOLE }  
var  
    a,b,hasil: real;  
    pilihan: char;  
begin  
    writeln('Pilihlah operasi berikut ini');  
    writeln('1. Perkalian');  
    writeln('2. Pembagian');  
    write('Pilihan Anda: ');  
    readln(pilihan);  
    if pilihan='1' then  
    begin  
        write('a = '); readln(a);  
        write('b = '); readln(b);  
        hasil:= a*b;  
        writeln('hasil = ',hasil:5:2);  
    end  
    else if pilihan='2' then  
    begin  
        write('a = '); readln(a);  
        write('b = '); readln(b);  
        if b=0 then  
        begin  
            writeln('Penyebut 0');  
            writeln('Hasil tidak terdefinisi');  
        end  
        else  
        begin  
            hasil:=a/b;  
            writeln('hasil = ', hasil:5:2);  
        end;  
    end  
    else  
        writeln('Pilihan Anda tidak ada');  
    readln;  
end.
```

### 2.4.1.2 Case - Of

Jika pilihan kondisi hanya sedikit, kita bisa menggunakan If. Namun jika pilihannya banyak, kita harus menggunakan Case - Of. Statement Case – Of digunakan untuk percabangan yang banyak, Sintaks penggunaannya adalah :

```
Case Variabel Kondisi Of  
Case – Label 1: [blok perintah1];  
Case – Label 2: [blok perintah2];  
Case – Label 3: [blok perintah3];
```

```
Case – Label 4: [blok perintah4];  
.....  
.....  
Case – Label n ; [blok perintahn] ;  
end;
```

```
program CobaCaseOf;  
uses  
    Windows, SysUtils;  
  
{ $R *.res }  
{ $APPTYPE CONSOLE }  
var  
    suhu, hasil: real;  
    pilihan: char;  
begin  
    writeln('Konversi Suhu');  
    writeln('=====');  
    writeln('1. Celcius ke Reamur');  
    writeln('2. Celcius ke Fahrenheit');  
    writeln('3. Reamur ke Fahrenheit');  
    writeln('Pilihan Anda : '); readln(pilihan);  
    case pilihan of  
        '1': begin  
            write('Suhu = '); readln(suhu);  
            hasil:= 4/5*suhu;  
            writeln('Suhu hasil konversi ', hasil:5:2 , 'derajad');  
            end;  
        '2': begin  
            write('Suhu = '); readln(suhu);  
            hasil:= 9/5*suhu+32;  
            writeln('Suhu hasil konversi ', hasil:5:2 , 'derajad');  
            end;  
        '3': begin  
            write('Suhu = '); readln(suhu);  
            hasil:= 9/4*suhu+32;  
            writeln('Suhu hasil konversi ', hasil:5:2 , 'derajad');  
            end;  
    else  
        writeln('Pilihan Anda tidak ada');  
    end;  
    readln;  
end.
```

## 2.4.2 Operasi Pengulangan

### 2.4.2.1 Repeat..Until

Statement repeat akan melaksanakan pengulangan proses terhadap blok perintah sampai suatu keadaan dinyatakan True (bernilai benar). Sintaks penggunaannya adalah sebagai berikut:

```
Repeat  
    [blok perintah]  
until kondisi
```

```
program CobaRepeatUntil;
uses
  Windows, SysUtils;

{$R *.res}
{$APPTYPE CONSOLE}
var
  luas, keliling, diameter: real;
  pilihan: char;
begin
  writeln('Menghitung Luas dan Keliling Lingkaran');
  writeln('=====');
  repeat
    write('Diameter = '); readln(diameter);
    luas:= 1/4*(22/7)*(diameter*diameter);
    keliling:= pi*diameter;
    writeln('Luas lingkaran : ', luas:5:2);
    writeln('Keliling lingkaran : ', keliling:5:2);
    write('Anda ingin menghitung lagi [Y/T] : '); readln(pilihan);
  until (pilihan = 't') or (pilihan = 'T');
  readln;
end.
```

#### 2.4.2.2 While..Do

Statement while digunakan untuk mengeksekusi sebuah blok secara berulang selama memenuhi kondisi tertentu. Sintaks penggunaannya adalah sebagai berikut:

```
While kondisi Do
begin
  [blok perintah];
end;
```

```
program CobaWhileDo;
uses
  Windows, SysUtils;

{$R *.res}
{$APPTYPE CONSOLE}
var
  a, b, fibbonaci, batas: integer;
  pilihan: char;
begin
  writeln('Mencari Bilangan Fibbonaci');
  writeln('=====');
  repeat
    write('Batas = '); readln(batas);
    if batas = 1 then
      writeln('Tidak ada bilangan fibbonaci kurang dari 1')
    else
      begin
        fibbonaci:= 1;
        a:= 0;
        b:= 1;
        while fibbonaci<batas do
          begin
            write(Fibbonaci, ' ');
            fibbonaci:= a+b;
            a:= b;
            b:= fibbonaci;
          end;
        end;
      end;
  until
    writeln;
```

```
write('Anda ingin lagi [Y/T] : ');
readln(pilihan);
until (pilihan = 't') or (pilihan = 'T');
end.
```

#### 2.4.2.3 For..Do

Statement for digunakan untuk mengeksekusi sebuah blok secara berulang dalam sebuah range tertentu. Sintaks penggunaannya adalah sebagai berikut:

**For** variabelawal **To** akhir **Do**  
[blok perintah]

```
program CobaForDo;
uses
  Windows, SysUtils;

{$R *.res}
{$APPTYPE CONSOLE}
var
  bilangan, i : integer;
  faktorial : extended;
  pilihan : char;
begin
  writeln('Program Menghitung Faktorial');
  writeln('=====');
  repeat
    write('Bilangan = '); readln(bilangan);
    if bilangan = 0 then faktorial:= 1
    else
      begin
        faktorial:= 1;
        for i:= 1 to bilangan do
          begin
            faktorial:= faktorial*i;
          end;
        end;
        writeln('Jumlah faktorial ',bilangan,' adalah ', faktorial:7:0);
        write('Ingin lagi [Y/T] : ');
        readln(pilihan);
      until (pilihan = 't') or (pilihan = 'T');
  readln;
end.
```

# Pembuatan Aplikasi

Aplikasi steganografi yang dibuat akan meyisipkan pesan pada format file BMP 24 bit. Format file BMP merupakan format file standar sistem operasi MS Windows 3.11/9x/NT dan IBM OS/2. Format file ini mendukung resolusi warna dari Monochrom hingga True Color (16,7 juta warna).

Format file BMP 24 bit menggunakan model warna RGB. Pada model warna RGB, warna yang ditampilkan di layar monitor disusun oleh tiga buah warna primer, yaitu Red (merah), Green (hijau), Blue (Biru). Pada model warna RGB setiap titik pada layar monitor berisi angka yang menunjukkan intensitas yang dipilih pada suatu tabel warna RGB. Jadi pada setiap titik dapat dipilih salah satu warna dari RGB.

Warna dari tabel RGB memiliki 3 buah warna kombinasi warna yaitu R,G, B, yang menentukan proporsi warna merah, hijau, biru dari warna tersebut. Warna di tabel yang dapat dipilih untuk mengisi warna pada sebuah pixel adalah  $256 \times 256 \times 256 = 16,7$  juta warna.

## 3.1 Metode Penyembunyian Pesan

Metode yang digunakan untuk penyembunyian pesan rahasia pada aplikasi ini adalah dengan cara menyisipkan pesan ke dalam bit rendah (*least significant bit*) pada data pixel yang menyusun file gambar BMP 24 bit tersebut.

Pada file gambar BMP 24 bit setiap pixel pada gambar terdiri dari susunan tiga warna yaitu merah, hijau, biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (1 byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Sebagai contoh file gambar BMP 24 bit dengan warna merah murni dalam format biner akan terlihat sebagai berikut:

00000000	00000000	11111111
00000000	00000000	11111111

Sedangkan untuk warna hijau murni dalam format biner akan terlihat sebagai berikut:

00000000	11111111	00000000
00000000	11111111	00000000

Sedangkan untuk warna biru murni dalam format biner akan terlihat sebagai berikut:

11111111	00000000	00000000
11111111	00000000	00000000

Dari uraian di atas dapat dilihat bahwa informasi dari warna biru berada pada bit pertama sampai bit delapan, dan informasi warna hijau berada pada bit sembilan sampai dengan bit 16, sedangkan informasi warna merah berada pada bit 17 sampai dengan bit 24.

Metode penyisipan LSB (*least significant bit*) ini adalah menyisipi pesan dengan cara mengganti bit ke 8, 16 dan 24 pada representasi biner file gambar dengan representasi biner pesan rahasia yang akan disembunyikan. Dengan demikian pada setiap pixel file gambar BMP 24 bit dapat disisipkan 3 bit pesan, misalnya terdapat data raster original file gambar adalah sebagai berikut:

00100111	11101001	11001000
00100111	11001000	11101001
11001000	00100111	11101001

Sedangkan representasi biner huruf A adalah 01000001, dengan menyisipkannya ke dalam pixel di atas maka akan dihasilkan

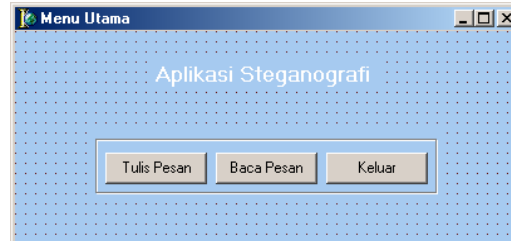
0010011 <b>0</b>	11101001	11001000
0010011 <b>0</b>	11001000	1110100 <b>0</b>
11001000	00100111	11101001

Terlihat pada bit kedelapan, enambelas dan 24 diganti dengan representasi biner huruf A, dan hanya tiga bit rendah yang berubah (cetak tebal), untuk penglihatan mata manusia sangatlah mustahil untuk dapat membedakan warna pada file gambar yang sudah diisi pesan rahasia jika dibandingkan dengan file gambar asli sebelum disisipi dengan pesan rahasia.

## 3.2 Menu Utama

Antar muka Menu Utama ini merupakan antar muka yang pertama kali muncul pada aplikasi. Pada antar muka Menu Utama ini terdapat tiga buah tombol yang menghubungkan ke proses selanjutnya yaitu tombol Tulis Pesan yang jika diklik akan masuk ke antar muka Tulis Pesan, tombol Baca Pesan yang jika diklik akan masuk ke antar muka Baca Pesan dan tombol Keluar yang jika diklik akan keluar dari aplikasi.

Tampilan rancangan antar muka form Menu Utama dapat dilihat pada gambar 3.1 di bawah ini.



**Gambar 3.1** Rancangan Form Menu Utama

Rincian masing-masing komponen yang ada pada form Menu Utama dapat dilihat pada tabel 3.1 di bawah ini.

**Tabel 3.1** Rincian komponen form Menu Utama

Komponen	Properti	Nilai
Form1	Caption	Menu Utama
	Color	clSkyBlue
	Name	frmTulisPesan
Button1	Caption	&Tulis Pesan
	Name	BtnTulisPesan
Button2	Caption	&Baca Pesan
	Name	BtnBaca Pesan
Button3	Caption	&Keluar
	Name	BtnKeluar
GroupBox1	Caption	
	Name	GroupButton

Setelah membuat rancangan form Menu Utama, selanjutnya kita dapat memasukkan program ke dalam form tersebut. Berikut beberapa langkah pemrograman form Menu Utama:

1. Klik ganda button Tulis Pesan, kemudian ketikkan program berikut:

```
procedure TfrmUtama.BtnTulisPesanClick(Sender: TObject);
begin
    frmTulisPesan.Show;
    with frmTulisPesan do
    begin
        ImageFile.Picture.Bitmap:=nil;
        MenuTulisPesan.Enabled:=False;
        MenuBacaPesan.Enabled:=True;
        MenuNew.Enabled:=True;
        MenuOpen.Enabled:=True;
        MenuSave.Enabled:=False;
        MenuCut.Enabled:=False;
        MenuCopy.Enabled:=False;
        MenuPaste.Enabled:=False;
        BtnTulisPesan.Enabled:=False;
        BtnBatal.Enabled:=False;
        BtnCekGambar.Enabled:=False;
        MemoTulisPesan.Visible:=False;
        MemoTulisPesan.Lines.Clear;
        EditMaks.Enabled:=False;
        EditMaks.Color:=clSilver;
        EditMaks.Clear;
        EditSisa.Enabled:=False;
        EditSisa.Color:=clSilver;
        EditSisa.Clear;
    end;
end;
```

2. Klik ganda button Baca Pesan, kemudian ketikkan program berikut:

```
procedure TfrmUtama.BtnBacaPesanClick(Sender: TObject);
begin
    frmBacaPesan.Show;
    with frmBacaPesan do
    begin
        ImageFile.Picture.Bitmap:=nil;
        MenuBacaPesan.Enabled:=False;
        MenuSave.Enabled:=False;
        MenuCut.Enabled:=False;
        MenuCopy.Enabled:=False;
        MenuPaste.Enabled:=False;
        BtnBacaPesan.Enabled:=False;
        BtnCekGambar.Enabled:=False;
        MemoBacaPesan.Visible:=False;
        MemoBacaPesan.Lines.Clear;
        EditMaks.Enabled:=False;
        EditMaks.Color:=clSilver;
        EditMaks.Clear;
        EditPakai.Enabled:=False;
        EditPakai.Color:=clSilver;
        EditPakai.Clear;
        EditSisa.Enabled:=False;
        EditSisa.Color:=clSilver;
        EditSisa.Clear;
    end;
end;
```

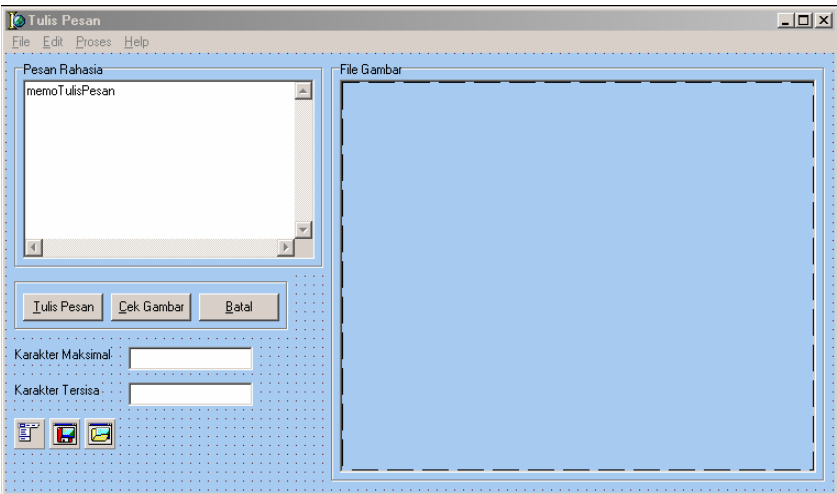
3. Klik ganda button Keluar, kemudian ketikkan program berikut:

```
procedure TfrmUtama.BtnKeluarClick(Sender: TObject);
var
  TombolMessageDlg : word;
begin
  TombolMessageDlg:=MessageDlg('Keluar dari Aplikasi ?', mtConfirmation,[mbYes,mbNo], 0);
  if TombolMessageDlg = mrYes then
    Application.Terminate;
end;
```

3.3 Form Tulis Pesan

Form Tulis Pesan merupakan antar muka untuk proses penyembunyian pesan. Dalam antar muka Tulis Pesan ini terdapat komponen Image yang dipakai untuk menempatkan file gambar sebagai file untuk disisipi pesan rahasia, komponen Memo yang dipakai untuk menempatkan penulisan pesan rahasia, tombol Batal digunakan untuk membatalkan penulisan pesan pada Memo, tombol Tulis Pesan yang jika diklik akan memulai proses penyembunyian pesan.

Tampilan rancangan antar muka form Tulis Pesan dapat dilihat pada gambar 3.2 di bawah ini.



Gambar 3.2 Rancangan Form Tulis Pesan

Rincian masing-masing komponen yang ada pada form Tulis Pesan dapat dilihat pada tabel 3.2.

Tabel 3.2 Rincian komponen form Tulis Pesan

Komponen	Properti	Nilai
Form1	Caption	Tulis Pesan
	Color	clSkyBlue
	Name	frmTulisPesan
Button1	Caption	&Tulis Pesan
	Name	BtnTulisPesan
Button2	Caption	&Cek Gambar
	Name	BtnCekGambar
Button3	Caption	&Batal
	Name	BtnBatal
MainMenu1	Name	MainMTPesan
MenuItem	Caption	&File
	Name	MenuFile
MenuItem	Caption	&Edit
	Name	MenuEdit
MenuItem	Caption	&Proses

	Name	MenuProses
MenuItem	Caption	&Help
	Name	MenuHelp
MenuItem	Caption	&New
	Name	MenuNew
MenuItem	Caption	&Open
	Name	MenuOpen
MenuItem	Caption	&Save
	Name	MenuSave
	ShortCut	Ctrl+S
MenuItem	Caption	E&xit
	Name	MenuExit
MenuItem	Caption	Cut
	Name	MenuCut
	ShortCut	Ctrl+X
MenuItem	Caption	Copy
	Name	MenuCopy
	ShortCut	Ctrl+C
MenuItem	Caption	Paste
	Name	MenuPaste
	ShortCut	Ctrl+V
MenuItem	Caption	Tulis Pesan
	Name	MenuTulisPesan
MenuItem	Caption	Baca Pesan
	Name	MenuBacaPesan
MenuItem	Caption	About
	Name	MenuAbout
Memo1	Name	MemoTulisPesan
	ScrollBar	ssBoth
Label1	Caption	Karakter Maksimal
	Name	Label1
Label2	Caption	Karakter Tersisa
	Name	Label2
Edit1	Name	EditMaks
	Text	
Edit2	Name	EditSisa
	Text	
GroupBox1	Caption	Pesan Rahasia
	Name	GroupPesan
GroupBox2	Caption	File Gambar
	Name	GroupImage
GroupBox3	Caption	
	Name	GroupButton
OpenPictureDialog1	Name	DialogBukaGambar
SavePictureDialog1	Name	DialogSimpanGambar
ScrollBar1	Name	ScrollBar1
Image1	Name	ImageFile

Setelah membuat rancangan form Tulis Pesan, selanjutnya kita dapat memasukkan program ke dalam form tersebut. Berikut beberapa langkah pemrograman form Tulis Pesan:

1. Ketikkan variabel yang akan digunakan pada form Tulis Pesan, seperti berikut:

```
Type DataPixel = Array [0..1000, 0..1000] of integer;
var
  frmTulisPesan: TfrmTulisPesan;
  MaxChar, InfoBaris, InfoKolom, tinggi, lebar, PanjangText : integer;
  NamaFile : string;
  hdc1 : hdc;
  Awal, Selesai : TDateTime;
```

2. Buatlah empat fungsi yaitu fungsi biner, biner16, pangkat, dan decimal, pada awal program seperti berikut:

```
function biner(angka:byte): string;
var
  a, sisa : integer;
  car : string[1];
  hasil : string[8];
begin
  hasil:='00000000';
  a:=0;
  if angka <> 0 then
    begin
      while angka <> 1 do
        begin
          sisa:=angka mod 2;
          str(sisa,car);
          hasil[8-a]:=car[1];
          angka:=trunc(angka/2);
          inc(a);
        end;
      str(angka,car);
      hasil[8-a]:=car[1];
    end;
    biner:=hasil;
  end;

function biner16(angka:word): string;
var
  a, sisa : integer;
  car : string[1];
  hasil : string[16];
begin
  hasil:='0000000000000000';
  a:=0;
  if angka <> 0 then
    begin
      while angka <> 1 do
        begin
          sisa:=angka mod 2;
          str(sisa, car);
          hasil[16-a]:=car[1];
          angka:=trunc(angka/2);
          inc(a);
        end;
      str(angka,car);
      hasil[16-a]:=car[1];
    end;
    biner16:=hasil;
  end;

function pangkat (x,y:integer):integer;
var
  i, hasil : integer;
begin
  hasil:=1;
  if y=0 then hasil:=1;
```

```
    for i:=1 to y do
      hasil:=hasil*x;
      pangkat:=hasil;
    end;

function decimal (bnr:string):integer;
var
  hasil, panjang, x:integer;
begin
  hasil:=0;
  panjang:=length(bnr);
  for x:=1 to panjang do
    if bnr[x]='1' then hasil:=hasil+pangkat(2,panjang-x);
  decimal:=hasil;
end;
```

3. Klik menu New, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuNewClick(Sender: TObject);
begin
  ImageFile.Picture.Bitmap:=nil;
  MenuOpen.Enabled:=True;
  MenuTulisPesan.Enabled:=False;
  MenuSave.Enabled:=False;
  BtnTulisPesan.Enabled:=False;
  BtnBatal.Enabled:=False;
  BtnCekGambar.Enabled:=False;
  MemoTulisPesan.Visible:=False;
  EditMaks.Clear;
  EditSisa.Clear;
end
```

4. Klik menu Open, kemudian ketikkan program berikut;

```
procedure TfrmTulisPesan.MenuOpenClick(Sender: TObject);
var
  NamaFile : string;
  tinggi,lebar,TextMaks : integer;
begin
  if DialogBukaGambar.Execute then
    begin
      NamaFile:=DialogBukaGambar.FileName;
      ImageFile.Picture.LoadFromFile(NamaFile);
      ImageFile.Height:=ImageFile.Picture.Height;
      ImageFile.Width:=ImageFile.Picture.Width;
      tinggi:=ImageFile.Height;
      lebar:=ImageFile.Width;
      if (lebar<24) or (tinggi<2) then
        begin
          MessageDlg('File Gambar harus lebar >=24 & tinggi>2, Cari File yang lain !',mtError,
[mbOK],0);
          ImageFile.Picture.Bitmap:=nil;
          MenuTulisPesan.Enabled:=False;
          MenuBacaPesan.Enabled:=True;
          MenuNew.Enabled:=True;
          MenuOpen.Enabled:=True;
          MenuSave.Enabled:=False;
          BtnTulisPesan.Enabled:=False;
          BtnBatal.Enabled:=False;
          BtnCekGambar.Enabled:=False;
          MemoTulisPesan.Visible:=False;
          MemoTulisPesan.Lines.Clear;
          EditMaks.Clear;
          EditSisa.Clear;
          exit;
        end;
    end;
end;
```

```
        end;  
        BtnTulisPesan.Enabled:=True;  
        BtnBatal.Enabled:=True;  
        BtnCekGambar.Enabled:=True;  
        MenuBacaPesan.Enabled:=False;  
        MemoTulisPesan.Visible:=True;  
        MemoTulisPesan.Lines.Clear;  
        TextMaks:=(lebar*(tinggi-1)*3) div 8;  
        MemoTulisPesan.MaxLength := TextMaks;  
        EditMaks.Text:=IntToStr(TextMaks);  
        EditSisa.Text:=IntToStr(TextMaks);  
    end;  
end;
```

5. Klik menu Save, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuSaveClick(Sender: TObject);  
var  
    BisaDisimpan: boolean;  
begin  
    DialogSimpanGambar.DefaultExt:='Bmp';  
    ImageFile.Height:=ImageFile.Picture.Height;  
    ImageFile.Width:=ImageFile.Picture.Width;  
    tinggi:= ImageFile.Height;  
    lebar:=ImageFile.Width;  
    if DialogSimpanGambar.Execute then  
        begin  
            NamaFile:=DialogSimpanGambar.FileName;  
            if not FileExists(DialogSimpanGambar.FileName) then  
                BisaDisimpan:=true  
            else  
                BisaDisimpan:=MessageDlg('File'+NamaFile+'sudah ada, tetap simpan?',  
mtConfirmation,[mbYes,mbNo],0)=mrYes;  
            if BisaDisimpan then  
                begin  
                    ImageFile.Picture.SaveToFile(NamaFile);  
                    BtnCekGambar.Enabled:=False;  
                    ImageFile.Picture.Bitmap:= nil;  
                    BtnBatal.Enabled:=False;  
                    BtnTulisPesan.Enabled:=False;  
                    MenuBacaPesan.Enabled:=True;  
                    MenuOpen.Enabled:=True;  
                    MenuSave.Enabled:=False;  
                end  
            else  
                MenuSave.Enabled:=True;  
            end;  
        end;  
    end;  
end;
```

6. Klik menu Exit, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuExitClick(Sender: TObject);  
var  
    TombolMessageDlg : word;  
begin  
    TombolMessageDlg:=MessageDlg('Keluar dari form Tulis Pesan?',  
mtConfirmation,[mbYes,mbNo], 0);  
    if TombolMessageDlg = mrYes then  
        frmTulisPesan.Close;  
    end;
```

7. Klik menu Cut, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuCutClick(Sender: TObject);  
begin  
    MemoTulisPesan.CutToClipboard;  
end;
```

8. Klik menu Copy, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuCopyClick(Sender: TObject);  
begin  
    MemoTulisPesan.CopyToClipboard;  
end;
```

9. Klik menu Paste, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuPasteClick(Sender: TObject);  
begin  
    MemoTulisPesan.PasteFromClipboard;  
end;
```

10. Klik menu Tulis Pesan, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuTulisPesanClick(Sender: TObject);  
begin  
    MenuTulisPesan.Enabled:=False;  
end;
```

11. Klik menu Baca Pesan, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.MenuBacaPesanClick(Sender: TObject);
var
    TombolMessageDlg: word;
begin
    TombolMessageDlg := MessageDlg('Berpindah ke form Baca Pesan?',
mtConfirmation,[mbYes,mbNo],0);
    if TombolMessageDlg = mrYes then
        begin
            frmBacaPesan.Show;
            with frmBacaPesan do
                begin
                    ImageFile.Picture.Bitmap:=nil;
                    MenuBacaPesan.Enabled:=False;
                    EditMaks.Clear;
                    MenuSave.Enabled:=False;
                    MenuCut.Enabled:=False;
                    MenuCopy.Enabled:=False;
                    MenuPaste.Enabled:=False;
                    BtnBacaPesan.Enabled:=False;
                    BtnCekGambar.Enabled:=False;
                    MemoBacaPesan.Lines.Clear;
                    MemoBacaPesan.Visible:=False;
                    EditMaks.Enabled:=False;
                    EditMaks.Color:=clSilver;
                    EditPakai.Enabled:=False;
                    EditPakai.Color:=clSilver;
                    EditSisa.Enabled:=False;
                    EditSisa.Color:=clSilver;
                    frmTulisPesan.Close;
                end;
            end;
        else
            frmTulisPesan.Show;
            MenuTulisPesan.Enabled:=False;
            MenuSave.Enabled:=False;
            MenuCut.Enabled:=False;
            MenuCopy.Enabled:=False;
            MenuPaste.Enabled:=False;
            BtnTulisPesan.Enabled:=False;
            BtnBatal.Enabled:=False;
            BtnCekGambar.Enabled:=False;
        end;
end;
```

12. Klik menu About, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuAboutClick(Sender: TObject);
begin
    frmAbout.ShowModal;
end;
```

13. Klik ganda button Cek Gambar, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.BtnCekGambarClick(Sender: TObject);
var
    i, merah, hijau, biru: integer;
    fleck, CekBiner, huruf1, huruf2, huruf3: string;
begin
    for i:=0 to 7 do
        begin
            merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
            hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
            biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
            CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+ (biner(merah))[8];
        end;
        huruf1:=chr(decimal(copy(CekBiner,1,8)));
        huruf2:=chr(decimal(copy(CekBiner,9,8)));
    end;
```

```
huruf3:=chr(decimal(copy(CekBiner,17,8)));
fleck:=huruf1+huruf2+huruf3;
if fleck = '@#$$' then
begin
    MessageDlg('File gambar sudah berisi pesan, cari file gambar lain !', mtError, [mbOK],0);
    ImageFile.Picture.Bitmap:=nil;
    MenuTulisPesan.Enabled:=False;
    MenuBacaPesan.Enabled:=True;
    MenuSave.Enabled:=False;
    BtnTulisPesan.Enabled:=False;
    BtnBatal.Enabled:=False;
    BtnCekGambar.Enabled:=False;
    MemoTulisPesan.Visible:=False;
    MemoTulisPesan.Lines.Clear;
    EditMaks.Clear;
    EditSisa.Clear;
end
else
begin
    MessageDlg('File gambar belum berisi pesan!',mtConfirmation,[mbOK],0);
end;
end;
```

14. Klik ganda button Tulis Pesan, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.BtnTulisPesanClick(Sender: TObject);
var
    i, merah, hijau, biru, MerahBaru, BiruBaru, HijauBaru, InfoKolom, InfoBaris, n, j, z, x: integer;
    fleck, BinerMerah, BinerHijau, BinerBiru, CekBiner, BinerPesan, huruf1, huruf2, huruf3,
    BinerFleck, BinerBaris, BinerKolom, pesan: string;
begin
    ImageFile.Height:=ImageFile.Picture.Height;
    ImageFile.Width:=ImageFile.Picture.Width;
    tinggi:=ImageFile.Height;
    lebar:=ImageFile.Width;
    if memoTulisPesan.Lines.Text="" then
    begin
        MessageDlg('Pesan belum diisi !',mtError, [mbOK], 0);
        exit;
    end
    else
    begin
        BtnTulisPesan.Enabled:=False;
        BtnBatal.Enabled:=False;
        BtnCekGambar.Enabled:=False;
        MenuBacaPesan.Enabled:=False;
        hdc1:=ImageFile.Canvas.Handle;

        //cek fleck
        for i:=0 to 7 do
        begin
            merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
            hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
            biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
            CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+ (biner(merah))[8];
        end;
        huruf1:=chr(decimal(copy(CekBiner,1,8)));
        huruf2:=chr(decimal(copy(CekBiner,9,8)));
        huruf3:=chr(decimal(copy(CekBiner,17,8)));
        fleck:=huruf1+huruf2+huruf3;
        if fleck = '@#$$' then
        begin
            MessageDlg('File gambar sudah berisi pesan, cari File yang lain!', mtError, [mbOK], 0);
            ImageFile.Picture.Bitmap:=nil;
            MenuTulisPesan.Enabled:=False;
            MenuBacaPesan.Enabled:=True;
            MenuSave.Enabled:=False;
```

```
    BtnTulisPesan.Enabled:=False;
    BtnBatal.Enabled:=False;
    BtnCekGambar.Enabled:=False;
    MemoTulisPesan.Visible:=False;
    MemoTulisPesan.Lines.Clear;
    EditMaks.Clear;
    EditSisa.Clear;
    exit;
end
else
    //sisip fleck
    Screen.Cursor:=CrHourGlass;
    fleck:='@#$';
    BinerFleck:='';
    n:=1;
    for i:=1 to 3 do
        BinerFleck:=BinerFleck+biner(ord(fleck[i]));
    z:=0;
    for i:=0 to 7 do
        begin
            inc(z);
            merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
            hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
            biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
            BinerMerah:=biner(merah);
            BinerHijau:=biner(hijau);
            BinerBiru:=biner(biru);
            delete(BinerMerah,8,1);
            delete(BinerHijau,8,1);
            delete(BinerBiru,8,1);
            BinerBiru:=BinerBiru+BinerFleck[n];
            BinerHijau:=BinerHijau+BinerFleck [n+1];
            BinerMerah:=BinerMerah+BinerFleck [n+2];
            n:=n+3;
            MerahBaru:=decimal(BinerMerah);
            HijauBaru:=decimal(BinerHijau);
            BiruBaru:=decimal(BinerBiru);
            setpixelV(hdc1,i,0,RGB(MerahBaru,HijauBaru, BiruBaru));
        end;
    //sisip_pesan
    pesan:=memoTulisPesan.Lines.Text;
    BinerPesan:='';
    PanjangText:=length(memoTulisPesan.Lines.Text);
    for i:=1 to PanjangText do
        begin
            BinerPesan:=BinerPesan+biner(ord(pesan[i]));
        end;
    n:=1;
    for j:=1 to ImageFile.Height-1 do
        begin
            for i:=0 to ImageFile.Width-1 do
                begin
                    inc(z);
                    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,j));
                    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,j));
                    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,j));
                    BinerMerah:=biner(merah);
                    BinerHijau:=biner(hijau);
                    BinerBiru:=biner(biru);
                    delete(BinerMerah,8,1);
                    delete(BinerHijau,8,1);
                    delete(BinerBiru,8,1);
                    BinerBiru:=BinerBiru+BinerPesan[n];
                    BinerHijau:=BinerHijau+BinerPesan[n+1];
```

```
        BinerMerah:=BinerMerah+BinerPesan[n+2];
        n:=n+3;
        MerahBaru:=decimal(BinerMerah);
        HijauBaru:=decimal(BinerHijau);
        BiruBaru:=decimal(BinerBiru);
        setpixelV(hdc1,i,j,RGB(MerahBaru,HijauBaru,BiruBaru));
        if n>length(BinerPesan) then
            break;
        end;
        if n>length(BinerPesan) then
            break;
        end;
//sisip baris
InfoBaris:=j;
BinerBaris:=biner16(InfoBaris);
n:=1;
for x:=8 to 15 do
begin
    inc(z);
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,x,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,x,0));
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,x,0));
    BinerHijau:=biner(hijau);
    BinerMerah:=biner(merah);
    delete(BinerHijau,8,1);
    delete(BinerMerah,8,1);
    BinerHijau:=BinerHijau+BinerBaris[n];
    BinerMerah:=BinerMerah+BinerBaris[n+1];
    HijauBaru:=decimal(BinerHijau);
    MerahBaru:=decimal(BinerMerah);
    n:=n+2;
    setpixelV(ImageFile.Canvas.Handle,x,0,RGB(MerahBaru,HijauBaru, biru));
end;
//sisip_kolom
InfoKolom:= i;
BinerKolom:=biner16(InfoKolom);
n:=1;
for x:=16 to 23 do
begin
    inc(z);
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,x,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,x,0));
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,x,0));
    BinerHijau:=biner(hijau);
    BinerMerah:=biner(merah);
    delete(BinerHijau,8,1);
    delete(BinerMerah,8,1);
    BinerHijau:=BinerHijau+BinerKolom[n];
    BinerMerah:=BinerMerah+BinerKolom[n+1];
    HijauBaru:=decimal(BinerHijau);
    MerahBaru:=decimal(BinerMerah);
    n:=n+2;
    setpixelV(ImageFile.Canvas.Handle,x,0,RGB(MerahBaru,HijauBaru,biro));
end;
Screen.Cursor:=crdefault;
ImageFile.Refresh;
MenuOpen.Enabled:=False;
MenuNew.Enabled:=False;
MenuBacaPesan.Enabled:=False;
MenuSave.Enabled:=True;
MemoTulisPesan.Lines.clear;
EditMaks.Clear;
EditSisa.Clear;
MemoTulisPesan.Visible:=False;
```

```
end;
```

15. Klik ganda button Batal, kemudian ketikkan program berikut:

```
procedure TfrmTulisPesan.BtnBatalClick(Sender: TObject);  
begin  
    ImageFile.Picture.Bitmap:=nil;  
    MenuTulisPesan.Enabled:=False;  
    MenuBacaPesan.Enabled:=True;  
    MemoTulisPesan.Lines.Clear;  
    MemoTulisPesan.Visible:=False;  
    EditMaks.Clear;  
    EditSisa.Clear;  
end;
```

16. Ketikkan program berikut, dengan terlebih dahulu mengatur event Key Down pada MemoTulisPesan Prosedur di bawah ini digunakan untuk mengetahui sisa karakter yang dapat ditampilkan oleh gambar. Ini merupakan langkah akhir dalam memasukkan program ke dalam form Tulis Pesan.

```
procedure TfrmTulisPesan.MemoTulisPesanKeyDown(Sender: TObject; var Key: Word; Shift:  
TShiftState);  
var  
    sisa, x : integer;  
begin  
    EditSisa.Text:=EditMaks.Text;  
    x:=length(MemoTulisPesan.Lines.Text)+1;  
    Sisa:=MemoTulisPesan.MaxLength-x;  
    EditSisa.Text:=IntToStr(sisa);  
end;
```

Pada Form Tulis Pesan tersebut setelah menulis pesan rahasia selanjutnya pengguna tinggal menekan tombol Tulis Pesan yang kemudian proses penyembunyian pesan dimulai. Dalam modul program ini terdapat 4 proses utama yaitu cek fleck, sisip fleck, sisip pesan dan sisip lokasi yang dilakukan secara berurutan.

Fleck adalah tanda di dalam file gambar berupa deretan 3 karakter yaitu @#\$. Sebelum disisipi pesan, file gambar diperiksa dahulu oleh program apakah terdapat fleck di dalamnya, jika program menemukan fleck maka di dalam file gambar tersebut juga terdapat pesan rahasia.

Pendeteksian fleck ini terdapat pada proses cek fleck di dalam modul program Tulis Pesan.

```
//cek fleck
for i:=0 to 7 do
begin
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
    CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+(biner(merah))[8];
end;
huruf1:=chr(decimal(copy(CekBiner,1,8)));
huruf2:=chr(decimal(copy(CekBiner,9,8)));
huruf3:=chr(decimal(copy(CekBiner,17,8)));
fleck:=huruf1+huruf2+huruf3;
if fleck='@#$$' then
begin
    MessageDlg('File gambar sudah berisi pesan, cari File yang lain!', mtError, [mbOK], 0);
    ImageFile.picture.Bitmap:=nil;
    MenuTulisPesan.Enabled:=False;
    MenuBacaPesan.Enabled:=True;
    MenuSave.Enabled:=False;
    BtnTulisPesan.Enabled:=False;
    BtnBatal.Enabled:=False;
    BtnCekGambar.Enabled:=False;
    MemoTulisPesan.Visible:=False;
    MemoTulisPesan.Lines.Clear;
    EditMaks.Clear;
    EditSisa.Clear;
    exit;
end
```

Jika program tidak menemukan fleck di dalam file gambar setelah proses cek fleck selesai maka program akan meneruskan proses selanjutnya yaitu proses sisip fleck.

Proses sisip fleck ini berfungsi untuk memberi tanda pada file gambar bahwa file gambar ini sudah mengandung pesan.

```
//sisip fleck
Screen.Cursor:=CrHourGlass;
fleck:='@#$$';
BinerFleck:='';
n:=1;
for i:=1 to 3 do
BinerFleck:= BinerFleck +biner(ord(fleck[i]));
z:=0;
for i:=0 to 7 do
begin
    inc(z);
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
    BinerMerah:=biner(merah);
    BinerHijau:=biner(hijau);
    BinerBiru:=biner(biru);
    delete(BinerMerah,8,1);
    delete(BinerHijau,8,1);
    delete(BinerBiru,8,1);
    BinerBiru:=BinerBiru+ BinerFleck [n];
    BinerHijau:=BinerHijau+ BinerFleck [n+1];
    BinerMerah:=BinerMerah+ BinerFleck [n+2];
    n:=n+3;
    MerahBaru:=decimal(BinerMerah);
    HijauBaru:=decimal(BinerHijau);
    BiruBaru:=decimal(BinerBiru);
    setpixelV(hdc1,i,0,RGB(MerahBaru,HijauBaru, BiruBaru));
end;
```

Setelah proses sisip fleck selesai, kemudian program akan melanjutkan proses selanjutnya yaitu proses sisip pesan.

```
//sisip pesan
pesan:=memoTulisPesan.Lines.Text;
BinerPesan:="";
PanjangText:=length(memoTulisPesan.Lines.Text);
for i:=1 to PanjangText do
begin
    BinerPesan:=BinerPesan+biner(ord(pesan[i]));
end;
n:=1;
for j:=1 to ImageFile.Height-1 do
begin
    for i:=0 to ImageFile.width-1 do
    begin
        inc(z);
        merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,j));
        hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,j));
        biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,j));
        BinerMerah:=biner(merah);
        BinerHijau:=biner(hijau);
        BinerBiru:=biner(biru);
        delete(BinerMerah,8,1);
        delete(BinerHijau,8,1);
        delete(BinerBiru,8,1);
        BinerBiru:=BinerBiru+BinerPesan[n];
        BinerHijau:=BinerHijau+BinerPesan[n+1];
        BinerMerah:=BinerMerah+BinerPesan[n+2];
        n:=n+3;
        MerahBaru:=decimal(BinerMerah);
        HijauBaru:=decimal(BinerHijau);
        BiruBaru:=decimal(BinerBiru);
        setpixelV(hdc1,i,j,RGB(MerahBaru,HijauBaru,BiruBaru));
        if n>length(BinerPesan) then
            break;
    end;
    if n>length(BinerPesan) then
        break;
end;
```

Kemudian setelah proses sisip pesan ini selesai, program akan melanjutkan proses terakhir dalam modul program Tulis Pesan yaitu proses sisip lokasi.

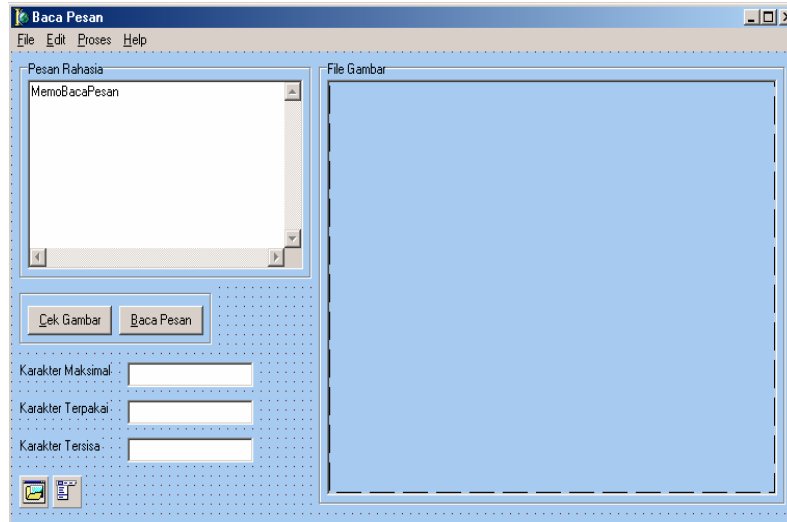
Proses sisip lokasi ini adalah proses memasukkan informasi tempat baris dan kolom terakhir dimana pesan rahasia disisipkan di dalam file gambar. Informasi lokasi ini berguna agar dalam proses pemisahan pesan rahasia dari file gambar, program bisa tepat dalam mengambil bit-bit biner pesan yang telah habis diambil oleh program.

```
//sisip baris
InfoBaris:=j;
BinerBaris:=biner16(InfoBaris);
n:=1;
for x:=8 to 15 do
begin
    inc(z);
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,x,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,x,0));
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,x,0));
    BinerHijau:=biner(hijau);
    BinerMerah:=biner(merah);
    delete(BinerHijau,8,1);
    delete(BinerMerah,8,1);
    BinerHijau:=BinerHijau+BinerBaris[n];
    BinerMerah:=BinerMerah+BinerBaris[n+1];
    HijauBaru:=decimal(BinerHijau);
    MerahBaru:=decimal(BinerMerah);
    n:=n+2;
    setpixelV(ImageFile.Canvas.Handle,x,0,RGB(MerahBaru,HijauBaru,biru));
end;
//sisip_kolom
InfoKolom:= i;
BinerKolom:=biner16(InfoKolom);
n:=1;
for x:=16 to 23 do
begin
    inc(z);
    biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,x,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,x,0));
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,x,0));
    BinerHijau:=biner(hijau);
    BinerMerah:=biner(merah);
    delete(BinerHijau,8,1);
    delete(BinerMerah,8,1);
    BinerHijau:=BinerHijau+BinerKolom[n];
    BinerMerah:=BinerMerah+ BinerKolom[n+1];
    HijauBaru:=decimal(BinerHijau);
    MerahBaru:=decimal(BinerMerah);
    n:=n+2;
    setpixelV(ImageFile.Canvas.Handle,x,0,RGB(MerahBaru,HijauBaru,biru));
end;
```

### 3.4 Form Baca Pesan

Form Baca Pesan ini merupakan antar muka untuk proses pemisahan pesan rahasia dari file gambar dan antar muka untuk membaca pesan rahasia. Pada antar muka Baca Pesan ini terdapat komponen Image yang dipakai untuk menempatkan file gambar yang sudah disisipi pesan rahasia, komponen Memo yang dipakai untuk menempatkan pesan rahasia yang akan dibaca, tombol Baca Pesan yang jika diklik akan memulai proses pemisahan pesan rahasia dari file gambar.

Tampilan rancangan antar muka form Baca Pesan dapat dilihat pada gambar 3.3 di bawah ini.



**Gambar 3.3** Rancangan Form Baca Pesan

Rincian masing-masing komponen yang ada pada form Baca Pesan dapat dilihat pada tabel 3.3 di bawah ini.

**Tabel 3.3** Rincian Komponen Form Baca Pesan

Komponen	Properti	Nilai
Form1	Caption	Baca Pesan
	Color	clSkyBlue
	Name	frmBacaPesan
Button1	Caption	&Cek Gambar
	Name	BtnCekGambar
Button2	Caption	&BacaPesan
	Name	BtnBacaPesan
MainMenu1	Name	MainMenuBacaPesan
MenuItem	Caption	&File
	Name	MenuFile
MenuItem	Caption	&Edit
	Name	MenuEdit
MenuItem	Caption	&Proses
	Name	MenuProses
MenuItem	Caption	&Help
	Name	MenuHelp
MenuItem	Caption	&New
	Name	MenuNew
MenuItem	Caption	&Open
	Name	MenuOpen
MenuItem	Caption	&Save
	Name	MenuSave
	ShortCut	Ctrl+S
MenuItem	Caption	E&xit
	Name	MenuExit
MenuItem	Caption	Cut
	Name	MenuCut
	ShortCut	Ctrl+X
MenuItem	Caption	Copy

	Name	MenuCopy
	ShortCut	Ctrl+C
MenuItem	Caption	Paste
	Name	MenuPaste
	ShortCut	Ctrl+V
MenuItem	Caption	Tulis Pesan
	Name	MenuTulisPesan
MenuItem	Caption	Baca Pesan
	Name	MenuBacaPesan
MenuItem	Caption	About
	Name	MenuAbout
Memo1	Name	MemoBacaPesan
	ReadOnly	True
	ScrollBars	ssBoth
Label1	Caption	Karakter Maksimal
	Name	Label1
Label2	Caption	Karakter Terpakai
	Name	Label2
Label3	Caption	Karakter Tersisa
	Name	Label3
Edit1	Name	EditMaks
	Text	
Edit2	Name	EditPakai
	Text	
Edit3	Name	EditSisa
	Text	
GroupBox1	Caption	Pesan Rahasia
	Name	GroupPesan
GroupBox2	Caption	File Gambar
	Name	GroupImage
GroupBox3	Caption	
	Name	GroupButton
OpenPictureDialog1	Name	DialogBukaGambar
ScrollBar1	Name	ScrollBar1
Image1	Name	ImageFile

Setelah membuat rancangan form Baca Pesan, selanjutnya kita dapat memasukkan program ke dalam form tersebut. Berikut beberapa langkah pemrograman form Baca Pesan:

1. Ketikkan variabel yang akan digunakan pada form Baca Pesan, seperti berikut:

```
var
    frmBacaPesan: TfrmBacaPesan;
    PanjangText: integer;
```

2. Buatlah empat fungsi yaitu fungsi biner, biner16, pangkat, dan decimal, pada awal program seperti berikut:

```
function biner(angka:byte): string;
var
    a, sisa : integer;
    car : string[1];
    hasil : string[8];
begin
    hasil:='00000000';
    a:=0;
    if angka <> 0 then
        begin
```

```
        while angka <> 1 do
        begin
            sisa:=angka mod 2;
            str(sisa,car);
            hasil[8-a]:=car[1];
            angka:=trunc(angka/2);
            inc(a);
        end;
        str(angka,car);
        hasil[8-a]:=car[1];
    end;
    biner:=hasil;
end;

function biner16(angka:word): string;
var
    a, sisa : integer;
    car : string[1];
    hasil : string[16];
begin
    hasil:='0000000000000000';
    a:=0;
    if angka <> 0 then
    begin
        while angka <> 1 do
        begin
            sisa:=angka mod 2;
            str(sisa, car);
            hasil[16-a]:=car[1];
            angka:=trunc(angka/2);
            inc(a);
        end;
        str(angka,car);
        hasil[16-a]:=car[1];
    end;
    biner16:=hasil;
end;

function pangkat (x,y:integer):integer;
var
    i, hasil : integer;
begin
    hasil:=1;
    if y=0 then hasil:=1;
    for i:=1 to y do
        hasil:=hasil*x;
    pangkat:=hasil;
end;

function decimal (bnr:string):integer;
var
    hasil, panjang, x:integer;
begin
    hasil:=0;
    panjang:=length(bnr);
    for x:=1 to panjang do
        if bnr[x]='1' then hasil:=hasil+pangkat(2,panjang-x);
    decimal:=hasil;
end;
```

3. Klik menu New, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuNewClick(Sender: TObject);
begin
    ImageFile.Picture.Bitmap:=nil;
    MenuOpen.Enabled:=True;
    MenuTulisPesan.Enabled:=False;
    MenuSave.Enabled:=False;
    BtnBacaPesan.Enabled:=False;
    BtnCekGambar.Enabled:=False;
    MemoBacaPesan.Lines.Clear;
    MemoBacaPesan.Visible:=False;
    EditMaks.Clear;
    EditPakai.Clear;
    EditSisa.Clear;
end;
```

4. Klik menu Open, kemudian ketikkan program berikut;

```
procedure TfrmBacaPesan.MenuOpenClick(Sender: TObject);
var
    NamaFile, pesan: string;
begin
    if DialogBukagambar.Execute then
    begin
        NamaFile := DialogBukaGambar.FileName;
        ImageFile.Picture.LoadFromFile(NamaFile);
        ImageFile.Height:=ImageFile.Picture.Height;
        ImageFile.Width:=ImageFile.Picture.Width;
        tinggi:=ImageFile.Height;
        lebar:=ImageFile.Width;
        if (lebar<24) or (tinggi<2) then
        begin
            MessageDlg('File gambar harus lebar>=24 & tinggi>2, Cari file yang lain!',
            mtError,[mbOK],0);
            ImageFile.Picture.Bitmap:=nil;
            MenuBacaPesan.Enabled:=False;
            MenuSave.Enabled:=False;
            BtnBacaPesan.Enabled:=False;
            BtnCekGambar.Enabled:=False;
            MemoBacaPesan.Visible:=False;
            MemoBacaPesan.Lines.Clear;
            exit;
        end;
        BtnBacaPesan.Enabled:=True;
        MenuSave.Enabled:=False;
        BtnCekGambar.Enabled:=True;
        MemoBacaPesan.Visible:=False;
        MemoBacaPesan.Lines.Clear;
        EditMaks.Clear;
        EditPakai.Clear;
        EditSisa.Clear;
    end;
end;
```

5. Klik menu Exit, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuExitClick(Sender: TObject);
var
    TombolMessageDlg: word;
begin
    TombolMessageDlg:=MessageDlg('Keluar dari form Baca
Pesan?',mtConfirmation,[mbYes,mbNo],0);
    if TombolMessageDlg = mrYes then
        frmBacaPesan.Close;
end;
```

6. Klik menu Cut, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuCutClick(Sender: TObject);
begin
    MemoBacaPesan.CutToClipboard;
end;
```

7. Klik menu Copy, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuCopyClick(Sender: TObject);
begin
    MemoBacaPesan.CopyToClipboard;
end;
```

8. Klik menu Paste, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuPasteClick(Sender: TObject);
begin
    MemoBacaPesan.PasteFromClipboard;
end;
```

9. Klik menu Tulis Pesan, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuTulisPesanClick(Sender: TObject);
var
    TombolMessageDlg: word;
begin
    TombolMessageDlg:=MessageDlg('Berpindah ke form Tulis
Pesan?',mtConfirmation,[mbYes,mbNo],0);
    if TombolMessageDlg= mrYes then
    begin
        frmTulisPesan.Show;
        with frmTulisPesan do
        begin
            ImageFile.Picture.Bitmap:=nil;
            MenuTulisPesan.Enabled:=False;
            MenuSave.Enabled:=False;
            MenuCut.Enabled:=False;
            MenuCopy.Enabled:=False;
            MenuPaste.Enabled:=False;
            BtnTulisPesan.Enabled:=False;
            BtnBatal.Enabled:=False;
            BtnCekGambar.Enabled:=False;
            MemoTulisPesan.Visible:=False;
            EditMaks.Enabled:=False;
            EditMaks.Color:=clSilver;
            EditSisa.Enabled:=False;
            EditSisa.Color:=clSilver;
            frmBacaPesan.Close;
        end;
    end
    else
        frmBacaPesan.Show;
        ImageFile.Picture.Bitmap:=nil;
        MenuBacaPesan.Enabled:=False;
        MenuSave.Enabled:=False;
        MenuCut.Enabled:=False;
        MenuCopy.Enabled:=False;
        MenuPaste.Enabled:=False;
        BtnBacaPesan.Enabled:=False;
        BtnCekGambar.Enabled:=False;
        EditMaks.Clear;
        EditPakai.Clear;
        EditSisa.Clear;
end;
```

10. Klik menu Baca Pesan, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuBacaPesanClick(Sender: TObject);  
begin  
    MenuBacaPesan.Enabled:=False;  
end;
```

11. Klik menu About, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.MenuAboutClick(Sender: TObject);  
begin  
    frmAbout.ShowModal;  
end;
```

12. Klik ganda button Cek Gambar, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.BtnCekGambarClick(Sender: TObject);  
var  
    i, merah, hijau, biru : integer;  
    fleck, CekBiner, huruf1, huruf2, huruf3 : string;  
begin  
    for i:=0 to 7 do  
    begin  
        merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));  
        hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));  
        biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));  
        CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+ (biner(merah))[8];  
    end;  
    huruf1:=chr(decimal(copy(CekBiner,1,8)));  
    huruf2:=chr(decimal(copy(CekBiner,9,8)));  
    huruf3:=chr(decimal(copy(CekBiner,17,8)));  
    fleck:=huruf1+huruf2+huruf3;  
    if fleck = '@#$$' then  
    begin  
        MessageDlg('File gambar berisi pesan',mtConfirmation,[mbOK],0);  
    end  
    else  
    begin  
        MessageDlg('File gambar tidak berisi pesan!',mtError,[mbOK],0);  
        ImageFile.Picture.Bitmap:=nil;  
        BtnBacaPesan.Enabled:=False;  
        BtnCekGambar.Enabled:=False;  
        EditMaks.Clear;  
        EditPakai.Clear;  
        EditSisa.Clear;  
    end;  
end;
```

13. Ini merupakan bagian akhir dari langkah dalam memasukkan program dalam form Baca Pesan.

Klik ganda button Baca Pesan, kemudian ketikkan program berikut:

```
procedure TfrmBacaPesan.BtnBacaPesanClick(Sender: TObject);  
var  
    i, merah, hijau, biru, n, j, z, InfoBaris, InfoKolom, total, lebar, tinggi, x, TextMaks, sisa, KonvNMaks,  
    KonvNPakai : integer;  
    fleck, BinerMerah, BinerHijau, BinerBiru, CekBiner, BinerPesan, huruf1, huruf2, huruf3, BinerBaris,  
    BinerKolom, pesan, AmbilNMaks, AmbilNPakai: string;  
begin  
    ImageFile.Height:=ImageFile.Picture.Height;  
    ImageFile.Width:= ImageFile.Picture.Width;  
    tinggi:=ImageFile.Height;  
    lebar:=ImageFile.Width;  
    BtnBacaPesan.Enabled:=True;  
    BtnCekGambar.Enabled:=True;  
    //cek fleck  
    for i:=0 to 7 do  
    begin  
        merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
```

```
hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));
CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+ (biner(merah))[8];
end;
huruf1:=chr(decimal(copy(CekBiner,1,8)));
huruf2:=chr(decimal(copy(CekBiner,9,8)));
huruf3:=chr(decimal(copy(CekBiner,17,8)));
fleck:=huruf1+huruf2+huruf3;
if fleck=' @#$' then
begin
    //ambil info baris
    BinerBaris:="";
    for i:=8 to 15 do
    begin
        merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
        hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
        BinerBaris:=BinerBaris+(biner(hijau))[8]+(biner(merah))[8];
    end;
    InfoBaris:=decimal(BinerBaris);
    //ambil info kolom
    BinerKolom:="";
    for i:=16 to 23 do
    begin
        merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
        hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
        BinerKolom:=BinerKolom+(biner(hijau))[8]+(biner(merah))[8];
    end;
    InfoKolom:=decimal(BinerKolom);
    //ambil pesan dari gambar
    BinerPesan:="";
    begin
        total:=(ImageFile.Width*(infobaris-1))+ InfoKolom +1;
        for j:=1 to ImageFile.Height - 1 do
        begin
            for i:=0 to ImageFile.Width-1 do
            begin
                inc(z);
                biru:=GetBValue(getpixel (ImageFile.Canvas.Handle,i,j));
                hijau:=GetGValue(getpixel (ImageFile.Canvas.Handle,i,j));
                merah:=GetRValue(getpixel (ImageFile.Canvas.Handle,i,j));
                BinerBiru:=biner(biru);
                BinerHijau:=biner(hijau);
                BinerMerah:=biner(merah);
                BinerPesan:=BinerPesan+copy(BinerBiru,8,1)+      copy(binerhijau,8,1)+
copy(binermerah,8,1);
                if (j=InfoBaris) and (i=InfoKolom) then
                break;
            end;
            if (j=InfoBaris) and (i=InfoKolom) then
            break;
        end;
        n:=1;
        pesan:="";
        for i:=1 to (round(length(BinerPesan)/8)) do
        begin
            inc(z);
            pesan:=pesan+chr(decimal(copy(BinerPesan,n,8)));
            n:=n+8;
        end;
        MenuBacaPesan.Enabled:=False;
        MenuTulisPesan.Enabled:=True;
        MemoBacaPesan.Visible:=True;
        MemoBacaPesan.Lines.Text:=pesan;
        Screen.Cursor:=crdefault;
```

```
end;  
TextMaks:=(lebar*(tinggi-1)*3) div 8;  
MemoBacaPesan.MaxLength := TextMaks;  
EditMaks.Text:=IntToStr(TextMaks);  
  
x:=length(MemoBacaPesan.Lines.Text);  
EditPakai.Text:=IntToStr(x);  
  
AmbilNMaks:=EditMaks.Text;  
AmbilNPakai:=EditPakai.Text;  
KonvNMaks:=StrToInt(AmbilNMaks);  
KonvNPakai:=StrToInt(AmbilNPakai);  
sis:=KonvNMaks-KonvNPakai;  
EditSisa.Text:=IntToStr(sisa);  
end  
else  
begin  
  
MessageDlg('File gambar tidak berisi pesan, cari file yang lain!',mtError,[mbOK],0);  
ImageFile.Picture.Bitmap:=nil;  
exit;  
MenuBacaPesan.Enabled:=False;  
MenuTulisPesan.Enabled:=True;  
end;  
end;  
end;
```

Dalam memisahkan pesan dari file gambar, Modul Program Baca Pesan menggunakan tiga proses utama yang dieksekusi secara berurutan yaitu cek fleck, ambil lokasi dan ambil pesan. Modul program Baca Pesan diawali oleh proses cek fleck yaitu proses yang memeriksa keberadaan fleck pada file gambar.

```
//cek fleck  
for i:=0 to 7 do  
begin  
merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));  
hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));  
biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,0));  
CekBiner:=CekBiner+(biner(biru))[8]+(biner(hijau))[8]+(biner(merah))[8];  
end;  
huruf1:=chr(decimal(copy(CekBiner,1,8)));  
huruf2:=chr(decimal(copy(CekBiner,9,8)));  
huruf3:=chr(decimal(copy(CekBiner,17,8)));  
fleck:=huruf1+huruf2+huruf3;  
if fleck = '@#$' then
```

Jika program tidak menemukan fleck pada file gambar dalam proses cek fleck, maka program akan berhenti dan akan menampilkan sebuah Message Dialog yang berisi bahwa file gambar tidak terdapat pesan.

Tetapi jika program menemukan fleck dalam file gambar, maka modul program Baca Pesan ke proses selanjutnya yaitu proses ambil lokasi.

Proses ambil lokasi ini adalah proses pengambilan informasi tempat baris dan kolom terakhir dimana pesan rahasia disisipkan di dalam file gambar. Informasi lokasi ini berguna agar dalam proses pemisahan pesan rahasia dari file gambar, program bisa tepat dalam mengambil bit-bit biner pesan, sehingga program bisa berhenti ketika bit-bit biner pesan telah habis diambil oleh program.

Informasi baris dan kolom terakhir ini berfungsi sebagai pembatas dalam *looping* pada proses ambil pesan, sehingga ketika sampai pada baris dan kolom terakhir proses ambil pesan bisa berhenti.

```
//ambil info baris
BinerBaris:="";
for i:=8 to 15 do
begin
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
    BinerBaris:=BinerBaris+(biner(hijau))[8]+(biner(merah))[8];
end;
InfoBaris:=decimal(BinerBaris);
//ambil info kolom
BinerKolom:="";
for i:=16 to 23 do
begin
    merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,0));
    hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,0));
    BinerKolom:=BinerKolom+(biner(hijau))[8]+(biner(merah))[8];
end;
InfoKolom:=decimal(BinerKolom);
```

Setelah proses ambil lokasi selesai dan mendapatkan hasil berupa informasi baris dan kolom terakhir penyisipan pesan maka program akan melanjutkan proses selanjutnya yaitu proses ambil pesan.

Proses ambil pesan ini adalah proses inti pemisahan pesan rahasia dari file gambar, sekaligus menampilkan pesan rahasia tersebut ke dalam form Baca Pesan sehingga pesan rahasia bisa dibaca oleh pengguna. *Looping* proses ini akan berhenti ketika sampai pada baris dan kolom terakhir penyisipan pesan yang informasinya didapat dari proses ambil lokasi, setelah *looping* berhenti kemudian program akan menampilkan pesan rahasia ke dalam form Baca Pesan.

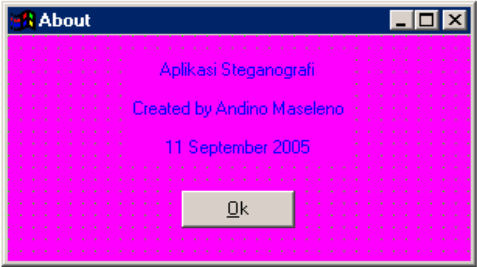
```
BinerPesan:="";
begin
    total:=(ImageFile.Width*(InfoBaris-1))+InfoKolom+1;
    for j:=1 to ImageFile.Height - 1 do
    begin
        for i:=0 to ImageFile.Width-1 do
        begin
            inc(z);
            biru:=GetBValue(getpixel(ImageFile.Canvas.Handle,i,j));
            hijau:=GetGValue(getpixel(ImageFile.Canvas.Handle,i,j));
            merah:=GetRValue(getpixel(ImageFile.Canvas.Handle,i,j));
            BinerBiru:=biner(biru);
            BinerHijau:=biner(hijau);
            BinerMerah:=biner(merah);

            BinerPesan:=BinerPesan+copy(BinerBiru,8,1)+copy(BinerHijau,8,1)+copy(BinerMerah,8,1);
            if (j=InfoBaris) and (i=InfoKolom) then
                break;
        end;
        if (j=InfoBaris) and (i=InfoKolom) then
            break;
        end;
        n:=1;
        pesan:="";
        for i:=1 to (round(length(BinerPesan)/8)) do
        begin
            inc(z);
            pesan:=pesan+chr(decimal(copy(BinerPesan,n,8)));
            n:=n+8;
        end;
        MenuBacaPesan.Enabled:=False;
        MenuTulisPesan.Enabled:=True;
        MemoBacaPesan.Visible:=True;
        MemoBacaPesan.Lines.Text:=pesan;
        Screen.Cursor:=crdefault;
```

### 3.5 Form About

Form About digunakan untuk menjelaskan mengenai aplikasi steganografi. form About merupakan bagian dari menu Help.

Tampilan rancangan antar muka form About dapat dilihat pada gambar 3.4 di bawah ini.



**Gambar 3.4** Rancangan Form About

Rincian masing-masing komponen yang ada pada form About dapat dilihat pada tabel 3.4 di bawah ini.

**Tabel 3.4** Rincian Komponen Form About

Komponen	Properti	Nilai
Form1	Caption	About
	BorderStyle	bsDialog
	Color	clYellow
	Name	frmAbout
Memo1	BorderStyle	bsNone
	Color	clFuchsia
	Font Color	clBlue
	Lines	Aplikasi Steganografi Created by Andino Maseleno 11 September 2005
Button1	Caption	&Ok
	Name	BtnOk

Berikut prosedur yang digunakan pada form about untuk menutup form tersebut.

```
procedure Tfrmabout.BtnOkClick(Sender: TObject);
begin
    frmAbout.Close;
end;
```

## DAFTAR PUSTAKA

- Fasnet, IT Center, 2004, *Modul Pelatihan Delphi*, IT Center Fasnet FT UGM, Yogyakarta.
- Maseleno, Andino, 2005, *Modul Pelatihan Delphi*, IT Community, Yogyakarta.
- Pramono, Djoko, 1999, *Mudah Menguasai Delphi 3.0 Jilid Satu*, PT Elex Media Komputindo, Jakarta.
- Pramono, Djoko, 2000, *Mudah Menguasai Delphi 3.0 Jilid Dua*, PT Elex Media Komputindo, Jakarta.
- Sellars, Duncan, An Introduction to Steganography,  
<http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html>
- Sukmawan, Budi, Steganografi, <http://students.ukdw.ac.id/~22033120/steganografi.html>
- USA Today, Terrorist instructions hidden online
- CNN.com, Bin Laden exploits technology to suit his needs
- Wired News, Bin Laden: Steganography Master?