

**PEMANFAATAN TCANVAS PADA DELPHI
UNTUK MEREPRESENTASIKAN JADWAL
(Studi Kasus : Jadwal Kuliah)**

Hendy Djaya Siswaja

Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI
Jl. Ir. H. Juanda 96 Bandung 40132

Abstrak

Jadwal dari suatu kegiatan terkadang sulit untuk direpresentasikan. Salah satunya adalah jadwal kuliah. Sering kali dalam proses penyusunan jadwal terjadi kesalahan yang diakibatkan kurang baiknya representasi dari jadwal yang ada. Representasi jadwal dalam sebuah *grid* kurang memuaskan karena tidak dapat dilihatnya sebaran kelas secara visual. Maka dari itu perlu cara lain untuk merepresentasikan jadwal. TCanvas adalah sebuah komponen pada Delphi yang dapat digunakan untuk menampilkan berkas-berkas grafik dan membuat grafik. TCanvas ini dapat dimanfaatkan untuk membuat representasi jadwal yang lebih baik.

Kata-kata kunci: jadwal, Delphi, TCanvas, grafik, baris, kolom.

1. PENDAHULUAN

Jadwal merupakan sesuatu yang sering digunakan dan dipakai oleh manusia. Hampir segala kegiatan manusia memiliki jadwal. Beberapa orang senang membuat jadwal untuk kegiatannya karena kegiatannya akan tersusun menjadi lebih rapi dan lebih mudah untuk mengerjakan kegiatan yang telah tersusun berdasarkan prioritas yang telah ditentukan sebelumnya serta membantu dalam membuat jadwal kegiatan baru tanpa harus mengganggu jadwal yang tersusun sebelumnya. Namun, jadwal adalah sesuatu yang tidak selalu mudah untuk direpresentasikan. Sering kali jadwal direpresentasikan dalam bentuk daftar (*list*) yang diurutkan berdasarkan hari atau waktu mulai atau prioritas lainnya.

Salah satu bentuk jadwal yang akan dijadikan studi kasus di sini adalah jadwal perkuliahan. Jadwal kuliah biasa terdiri dari mata kuliah, kelas, hari, ruang, dan waktu. Jika jadwal perkuliahan direpresentasikan dalam bentuk daftar, maka jadwal

tersebut akan sangat sulit untuk digunakan. Sebagai contoh, jika pihak akademik akan menambahkan jadwal kuliah baru maka pihak akademik sebelumnya perlu memeriksa terlebih dahulu hari, ruang, dan waktu yang masih tersedia untuk mencegah terjadinya bentrok antar jadwal. Dapat dibayangkan sulitnya untuk mencari slot jadwal yang masih tersedia hanya untuk menambahkan satu jadwal kuliah baru. Faktor lainnya adalah ketika pihak akademik ingin melihat sebaran jadwal yang telah tersusun untuk menentukan apakah jadwal yang telah tersusun cukup memiliki sebaran yang baik atau tidak. Jika sebarannya kurang baik, mungkin perlu ada penyesuaian pada jadwal kuliah yang telah ada. Hal seperti tersebut tidak mungkin dapat dilihat dalam bentuk daftar. Perlu dibuat bentuk lain yang dapat memenuhi kebutuhan tersebut.

2. KOMPONEN TCANVAS

TCanvas adalah sebuah komponen non-visual yang digunakan dimanapun dalam Delphi dimana proses menggambar dibutuhkan atau dimungkinkan dan membuat proses menggambar sebuah grafik menjadi lebih mudah dan tingkat kegagalannya minim. Hampir semua komponen visual pada Delphi memiliki atau dapat menggunakan TCanvas.

2.1 TPen

TPen adalah komponen non-visual yang digunakan untuk menggambar garis atau memberi outline pada sebuah bentuk dalam gambar. TPen memiliki beberapa *properties* didalamnya yakni :

1. Color, digunakan untuk menentukan warna dari TPen.
2. Mode, digunakan untuk menentukan bagaimana warna yang berasal dari TPen berinteraksi dengan warna pada TCanvas.
3. Style, digunakan untuk menentukan pola pembuatan garis oleh TPen.
4. Width, digunakan untuk menentukan lebar atau ketebalan dari TPen.

2.2 LineTo

LineTo adalah fungsi yang terdapat dalam TCanvas yang digunakan untuk membuat sebuah garis dari PenPos (posisi TPen) sampai dengan suatu titik tertentu yang direpresentasikan dengan koordinat X dan Y. Fungsi LineTo ini biasanya dituliskan sebagai berikut, LineTo(X,Y);

2.2.1 MoveTo

MoveTo adalah fungsi yang terdapat dalam TCanvas yang digunakan untuk memindahkan posisi dari TPen dari satu titik ke titik lain. Titik disini direpresentasikan dengan koordinat X dan Y. MoveTo berperan dalam menentukan PenPos sehingga biasanya MoveTo dan LineTo digunakan bersamaan. Fungsi MoveTo ini biasanya dituliskan sebagai berikut, MoveTo(X,Y);

2.4 TextOut

TextOut adalah fungsi dalam TCanvas yang digunakan untuk menampilkan sebuah string yang diinginkan pada gambar. TextOut juga membutuhkan koordinat X dan Y sebagai patokan posisi untuk menampilkan string tertentu. Fungsi TextOut biasanya dituliskan sebagai berikut, TextOut(X,Y,Text);

2.5 Pixels

Pixels adalah salah satu fungsi dalam TCanvas. Fungsi Pixels ini mempunyai dua jenis fungsi yakni, untuk membaca warna yang ada pada titik tertentu dan untuk memberikan warna tertentu pada titik tertentu. Namun penggunaan fungsi Pixels untuk memberikan warna membutuhkan waktu yang cukup lama karena pemberian warna dilakukan titik per titik. Fungsi Pixels biasanya dituliskan sebagai berikut, Pixels[X,Y];

2.6 Draw

Draw adalah fungsi dalam TCanvas yang digunakan untuk menggambarkan grafik yang telah ada sebelumnya pada TCanvas. Grafik ini dapat berupa sebuah file grafik yang dibuat dengan bantuan perangkat lunak grafik lain. Tipe grafik yang dapat diterima oleh fungsi Draw ini adalah Bitmap, Icon, atau metafiles. Fungsi Draw ini biasanya dituliskan sebagai berikut, Draw(X,Y,Graphics);

3. IMPLEMENTASI TCANVAS UNTUK MEREPRESENTASIKAN JADWAL KULIAH

Tahap awal adalah pemilihan komponen visual yang akan digunakan bersama dengan TCanvas untuk memvisualkan jadwal kuliah. TImage adalah salah satu komponen visual yang cocok untuk tujuan tersebut karena TImage dapat diletakkan

pada komponen visual lainnya. Beberapa TImage akan digunakan dalam implementasi ini karena jadwal kuliah akan dikelompokkan berdasarkan hari.

3.1 Inisialisasi Gambar Latar

TImage sebagai komponen visual secara default tidak memiliki gambar latar maupun warna dasar. Tanpa gambar latar, maka TCanvas tidak dapat digunakan pada TImage. Berikut adalah gambar dimana TImage digunakan sebagai komponen visual pada Delphi.



Gambar 1.1. Komponen TImage pada Design-Time

Untuk membuat gambar latar maka kita akan membuat sebuah *procedure* dimana gambar latar akan dibangun pada saat *run-time*. Berikut kode program dari *procedure* :

```
Procedure latar(img: TImage);  
var gbr : TBitmap;  
begin  
    gbr := nil;  
    try  
        gbr := TBitmap.Create;  
        gbr.Height := img.Height;  
        gbr.Width := img.Width;  
        img.Picture.Bitmap := gbr;  
    finally
```

```

        gbr.Free;
    end;
end;

```

Procedure di atas akan membuat sebuah grafik dengan tipe bitmap dan akan di-load ke dalam TImage. Selain berfungsi untuk membuat gambar latar, procedure ini juga dapat digunakan sebagai fungsi untuk membersihkan TImage dan TCanvas ketika akan membuat gambar jadwal yang baru. Berikut adalah gambar ketika procedure telah dibuat dan dipanggil pada events OnFormActivate.



Gambar 1.2. Aplikasi setelah menggunakan procedure latar

3.2 Pembuatan Kolom-Kolom Jadwal

Langkah berikutnya adalah membuat kolom-kolom jadwal yang berorientasi pada jam dan ruang. Kolom-kolom ini akan membantu dalam memvisualisasikan jadwal yang ada. Untuk membuat kolom-kolom ini, akan dibuat sebuah procedure lagi yakni, procedure buat_kolom. Isi dari procedure tersebut adalah sebagai berikut:

```

procedure buat_kolom(img: TImage);
var j,i : integer;
begin
    img.Canvas.Pen.Width := 1; //inisialisasi lebar TPen
    j := 60; //besaran yang menyatakan panjang dari tiap kolom

```

```
//looping dibawah ini akan membuat garis-garis vertikal dengan
//Jarak masing-masing sebesar 60 pixels dan looping dilakukan
//sebanyak waktu (jam) perkuliahan yang mungkin ada.
for i := 7 to 21 do
    begin
        img.Canvas.TextOut(j,0,inttostr(i));
        img.Canvas.MoveTo(j,20);
        img.Canvas.LineTo(j,320);
        inc(j,60);
    end;

j := 20; //besaran untuk menyatakan lebar dari tiap kolom

//looping dibawah ini akan membuat garis-garis horisontal
dengan
//Jarak masing-masing sebesar 20 pixels dan looping dilakukan
//sebanyak ruang perkuliahan yang mungkin ada.
for i := 0 to 15 do
    begin
        img.Canvas.MoveTo(60,j);
        img.Canvas.LineTo(900,j);
        inc(j,20);
    end;
//dikarenakan nama ruang yang berbeda-beda dan tidak ada pola
yang
//dapat dijadikan patokan, maka nama masing-masing ruang
//perkuliahan dibuat secara manual. Tentunya dengan
pengembangan
//lebih lanjut, ruangan dapat diambil dari database, tidak
statis.
img.Canvas.TextOut(10,24,'R-1204');
img.Canvas.TextOut(10,44,'R-1205');
img.Canvas.TextOut(10,64,'R-1207');
img.Canvas.TextOut(10,84,'R-1208');
img.Canvas.TextOut(10,104,'R-1209');
img.Canvas.TextOut(10,124,'R-1210');
img.Canvas.TextOut(10,144,'R-1211');
img.Canvas.TextOut(10,164,'R-1212');
img.Canvas.TextOut(10,184,'R-1304');
img.Canvas.TextOut(10,204,'R-1305');
```

```

img.Canvas.TextOut(10,224,'R-1307');
img.Canvas.TextOut(10,244,'R-1308');
img.Canvas.TextOut(10,264,'R-1309');
img.Canvas.TextOut(10,284,'R-1310');
img.Canvas.TextOut(10,304,'R-1400');
end;

```

Procedure tersebut akan menghasilkan kolom-kolom jadwal seperti pada gambar berikut ini :

Senin	Selasa	Rabu	Kamis	Jumat	Sabtu	Daftar mata kuliah				
	7	8	9	10	11	12	13	14	15	16
R-1204										
R-1205										
R-1207										
R-1208										
R-1209										
R-1210										
R-1211										
R-1212										
R-1304										
R-1305										
R-1307										
R-1308										
R-1309										
R-1310										
R-1400										

Gambar 1.3. Kolom-kolom jadwal pada TImage dengan berorientasi jam dan ruang

3.3 Menggambarkan Jadwal

Langkah terakhir adalah mengambil data dari database dan menggambarkannya ke TImage dengan TCanvas. Langkah ini dimulai dengan mengambil data dari database kemudian menentukan koordinat yang tepat untuk menggambar dengan cara melihat ruangan yang digunakan dan jam mulai dan selesai dari mata kuliah tertentu. Setelah mendapatkan koordinat yang tepat, jadwal dapat digambarkan ke dalam TImage. Kita juga dapat menambahkan pernik-pernik tambahan seperti warna untuk mata kuliah tahun pertama, kedua, dan seterusnya atau menentukan warna yang akan dipakai jika terjadi bentrok. Untuk menggambarkan jadwal ini, akan dibuat dua buah procedure yakni, procedure ambil_jadwal dan gambar_jadwal.

Berikut adalah isi dari procedure ambil_jadwal dan gambar_jadwal :

```

procedure gambar_jadwal(img: TImage; sem,x1,y1,x2,y2 : integer);
var i : integer;
    wrn : TBitmap;
begin
wrn := TBitmap.Create;
for i := x1 to x2 do
    begin
        //Jika ada sudah pernah digambar oleh jadwal lain
        if (img.Canvas.Pixels[i,y1+1]<>clWhite)
        and (img.Canvas.Pixels[i,y1+1]<>clBlack) then
            begin
                wrn.LoadFromFile('merah.bmp');
                img.Canvas.Draw(i,y1,wrn);
            end
        else
            begin
                //untuk semester yang berbeda-beda
                if (sem = 1) or (sem = 2) then
                    wrn.LoadFromFile('biru.bmp')
                else if (sem = 3) or (sem = 4) then
                    wrn.LoadFromFile('hijau.bmp')
                else if (sem = 5) or (sem = 6) then
                    wrn.LoadFromFile('kuning.bmp')
                else if (sem = 7) or (sem = 8) then
                    wrn.LoadFromFile('ungu.bmp');
                img.Canvas.Draw(i,y1,wrn);
            end;
        end;
    end;
end;

procedure ambil_jadwal(hari: integer);
var posisi_ruang, selisih_mulai_selesai, selisih_posisi: integer;
    img: TImage;
    selisih: TTime;
    ax,ay: integer;
begin
img:= nil;
posisi_ruang := 0;
ax:= 60; //panjang kolom
ay:= 20; //lebar kolom
case hari of

```

```

2 : img := img_senin;
3 : img := img_selasa;
4 : img := img_rabu;
5 : img := img_kamis;
6 : img := img_jumat;
7 : img := img_sabtu;
end;
Table1.Open; //table yang berisi data jadwal.
buat_grid(img);
while not table1.eof do
    begin
        with table1 do begin
            selisih:=fieldbyname('selesai').Value-
fieldbyname('mulai').Value;
            selisih_mulai_selesai := HourOf(selisih)*60+MinuteOf(selisih);
            selisih := fieldbyname('mulai').Value - strtotime('07:00:00');
            selisih_posisi := HourOf(selisih)*60+MinuteOf(selisih);
            case fieldbyname('kd_ruang').AsInteger of
                1204 : posisi_ruang := 20;
                1205 : posisi_ruang := 40;
                1207 : posisi_ruang := 60;
                1208 : posisi_ruang := 80;
                1209 : posisi_ruang := 100;
                1210 : posisi_ruang := 120;
                1211 : posisi_ruang := 140;
                1212 : posisi_ruang := 160;
                1304 : posisi_ruang := 180;
                1305 : posisi_ruang := 200;
                1307 : posisi_ruang := 220;
                1308 : posisi_ruang := 240;
                1309 : posisi_ruang := 260;
                1310 : posisi_ruang := 280;
                1400 : posisi_ruang := 300;
            end;
            gambar_jadwal(img,fieldbyname('semester').value,ax+selisih_posisi,pos
isi_ruang,ax+selisih_posisi+selisih_mulai_selesai,ay+posisi_ruang);

            img.Canvas.TextOut(ax+selisih_posisi+5,posisi_ruang+3,fieldbyname('Ko
de').value+'/'+fieldbyname('kelas').Value);
        end;
    next;
img.Repaint;

```

end;
end;
end;

Hasil dari kedua procedure tersebut diatas adalah jadwal akan tergambar pada kolom-kolom jadwal dengan warna yang berbeda-beda, seperti pada gambar berikut ini:

Senin	Selasa	Rabu	Kamis	Jumat	Sabtu	Daftar mata kuliah				
	7	8	9	10	11	12	13	14	15	16
R-1204	SI528/A	SI208/A	SI723/C							SI723/D
R-1205										
R-1207										
R-1208						SI621/A				SI228/A
R-1209	SI226/A	SI226/B						SI223/A		SI223/B
R-1210										
R-1211										
R-1212										
R-1304										
R-1305										
R-1307			SI221/C		SI230/A					
R-1308				SI721/A	SI221/A	SI221/B				KU206/A
R-1309		SI526/A	SI526/B					SI727/A		
R-1310	IF225/A	IF221/A		IF221/B				IF121/A		IF121/B
R-1400										

Gambar 1.4. Gambar jadwal kuliah yang tersedia

4. KESIMPULAN

TCanvas adalah komponen yang sangat berperan dalam pembuatan program visual dengan Delphi dan representasi jadwal kuliah dengan menggunakan TCanvas menjadi lebih baik dibandingkan dengan hanya menggunakan daftar. Dengan TCanvas, sebaran jadwal dapat terlihat dengan baik, begitu juga slot jadwal kosong untuk mengisi jadwal baru.

Masih banyak lagi yang dapat dikembangkan dengan TCanvas ini. Sebagai contoh, membuat jam analog, image editor sederhana, image blender, dan lain sebagainya. Penulis menyadari bahwa implementasi yang telah dibuat di atas masih jauh dari sempurna terutama masih banyaknya parameter-parameter yang dibuat secara statis dalam program. Namun penulis berharap, konsep yang dituangkan disini dapat menjadi inspirasi untuk mengembangkan sesuatu yang bahkan jauh lebih baik lagi.

5. Daftar Pustaka

1. Cantu, Marco, *Mastering Delphi 6*, Sybex Books, 2001.
2. Lischer, Ray, *Delphi In A Nutshell*, O'Reilly, 2000.
3. Telxelra, Steve and Pachecom, Xavler, *Borland Delphi 6 Developer's Guide*, Sams Publishing, 2002.