

Program Bantu Penyusunan Jadwal Kuliah

Antisipasi Jadwal Yang Tumpang Tindih

Katon Wijana
Program Studi Sistem Informasi
Universitas Kristen Duta Wacana
Yogyakarta, Indonesia
katony@staff.ukdw.ac.id

Abstrak— Menyusun jadwal kuliah memerlukan kecermatan agar tidak terjadi jadwal yang tumpang tindih (*overlap*). Tumpang tindih dapat terjadi pada dosen pengajar, mahasiswa peserta kuliah maupun ruang kuliah. Perancangan basis data dan pemanfaatan query SQL dapat membantu untuk mengatasi masalah penjadwalan ini.

Keywords—*overlap; jadwal; perancangan; basisdata; query*

I. PENDAHULUAN

Dalam penyusunan jadwal kuliah, penyusun harus memperhatikan dengan cermat jadwal masing-masing entitas yang terlibat, dalam hal ini dosen, mahasiswa dan ruang kuliah supaya tidak terjadi tumpang tindih jadwal (*overlap*).

Dalam memberikan jadwal mengajar bagi dosen yang harus diperhatikan adalah bahwa pada waktu yang sama dosen tidak bisa mengajar pada jadwal kuliah yang berbeda.

Dalam memberikan jadwal kuliah bagi mahasiswa, yang harus diperhatikan adalah bahwa jadwal mata kuliah-mata kuliah wajib bagi mahasiswa yang bersangkutan tidak saling tumpang tindih. Jadwal kuliah harus dibuat supaya mahasiswa dapat mengambil jadwal yang sesuai dengan porsi yang seharusnya.

Dalam memberikan jadwal pemakaian ruang kuliah tidak mungkin pada waktu yang sama dipergunakan oleh jadwal kuliah yang berbeda.

Hal terpenting dalam pembuatan program bantu penyusunan jadwal kuliah ini adalah perancangan basis data. Tiap-tiap entitas yang ada harus mempunyai atribut yang dibutuhkan untuk mendeteksi masalah tumpang tindih jadwal tersebut.

Hal yang tidak kalah pentingnya adalah pemanfaatan query untuk memeriksa kemungkinan adanya jadwal yang bertabrakan berdasarkan entitas-entitas yang ada.

II. TINJAUAN PUSTAKA

2.7. *Desain Basis Data*

Perancangan basis data merupakan aktivitas yang berfokus pada desain basis data yang akan digunakan untuk menyimpan dan mengatur data pemakai. Sebuah basis data yang memenuhi kebutuhan semua pemakai bukan hanya diperlukan tetapi strukturnya harus dirancang dengan hati-hati.

Data merupakan salah satu aset yang paling berharga, data konsumen, pemasok, karyawan, pemesanan dan transaksi semuanya merupakan bagian yang sangat vital untuk keberadaan sebuah perusahaan.

Dengan desain struktur data yang baik, kita akan dapat menggunakan perintah Query SQL sederhana untuk mendapatkan informasi yang dibutuhkan. [1]

2.8. *Structured Query Language*

Perintah SELECT menentukan kolom-kolom data yang akan ditampilkan, kriteria data yang diinginkan, dan urutan data yang dihasilkan. Selanjutnya perintah SELECT dapat dikelompokkan maupun digabungkan dengan tabel lain. Komponen perintah SELECT adalah seperti yang dijelaskan pada Tabel I. di bawah ini:

TABLE XIV. KOMPONEN PERINTAH SQL SELECT

Komponen	Keterangan
SELECT	Menentukan kolom-kolom yang dimaksud
FROM	Menentukan nama tabel yang dimaksud
WHERE	Menentukan kriteria data yang dimaksud
GROUP BY	Untuk mengelompokkan baris data
HAVING	Untuk menentukan kriteria query agregasi
ORDER BY	Untuk mengurutkan baris berdasarkan kolom

Perintah INSERT digunakan untuk menambah baris data pada tabel. Pada perintah INSERT ditentukan tabel mana yang akan ditambah barisnya, kolom apa saja yang hendak diberi

Prosiding
ANNUAL RESEARCH SEMINAR 2016
 6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI

http://ars.ilkom.unsri.ac.id

nilai dan dari mana sumber data diperoleh. Komponen perintah INSERT adalah seperti yang dijelaskan pada Tabel 2. di bawah ini:

TABLE XV. KOMPONEN PERINTAH SQL INSERT

Komponen	Keterangan
INSERT INTO	Menentukan tabel yang akan ditambah baris
daftar kolom	Menentukan nama-nama kolom yang akan diberi nilai
WHERE	Menentukan kriteria data yang dimaksud
VALUES	Menentukan nilai yang hendak diberikan
SELECT	Menentukan tabel sumber data yang hendak ditambahkan.

Perintah DELETE memungkinkan kita untuk menghapus baris data. Perintah ini menjelaskan tabel apa yang barisnya akan dihapus dan kriteria baris yang dimaksud. Komponen perintah DELETE adalah seperti yang dijelaskan pada Tabel 3. di bawah ini: [2]

TABLE XVI. KOMPONEN PERINTAH SQL DELETE

Komponen	Keterangan
DELETE FROM	Menentukan tabel yang akan dihapus
WHERE	Menentukan kriteria data yang dimaksud

2.9. Interaksi Manusia dan Komputer

Dalam buku *System Analysis and Design* yang ditulis oleh Kendall & Kendall mengatakan bahwa ineteraksi dengan pemakai yang minimal merupakan hal yang sangat penting dalam perancangan antarmuka grafis. Penekanan tombol keyboard merupakan bagian yang paling membuat jalannya sistem komputer menjadi lambat, oleh karena itu ada 8 petunjuk yang harus diperhatikan, yaitu :

1. Yang dimasukkan cukup kodenya saja, jangan semua teks yang ditampilkan pada layar.
2. Hanya memasukkan data yang belum tersimpan pada file atau tabel.
3. Sediakan karakter pengatur format data yang dibutuhkan tanpa pemakai menuliskannya.
4. Gunakan nilai default untuk field pada layar masukan.
5. Mendesain suatu penyaringan data yang mirip dengan data yang dikehendaki oleh pemakai.
6. Sediakan ketikan (*keystrokes*) yang dapat memunculkan menu pilihan.
7. Gunakan *Radio button* dan *Drop down list* untuk mengatur tampilan menu yang sesuai.
8. Atur posisi cursor agar berada pada tempat yang perlu untuk diakses berikutnya. [3]

2.10. Pemrograman C#

.NET Framework berisi sekumpulan *library* (pustaka) berupa kelas yang diatur dan dikelompokkan dalam bentuk hirarki *namespace*. *Class Library* dalam .NET Framework dibagi ke dalam dua kelompok : *Base Class Library* (BCL) dan *Framework Class Library* (FCL).

Framework Class Library (FCL) merupakan *superset* dari BCL dan mengacu ke seluruh *library* kelas yang terdapt di dalam .NET Class Library. FCL berisi sekumpulan kelas untuk pembuatan program yang melibatkan Windows Form (WinForm), ADO.NET, ASP.NET, *Language Integrated Query* (LINQ), *Windows Presentation Foundation* (WPF), *Windows Communication Foundation* (WCF) dan lain-lain. [4]

Antarmuka grafis memungkinkan pemakai berinteraksi dengan program secara visual. Sebuah antarmuka grafis didukung oleh kontrol-kontrol antarmuka grafis yang membuat pemakai dapat menghemat waktu karena tidak perlu untuk menghafalkan perintah-perintah apa yang harus diberikan melainkan tinggal memilih menu-menu dalam bentuk grafis. [5].

III. PERANCANGAN

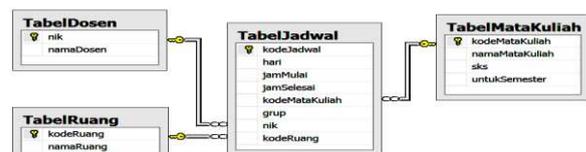
Pada penelitian ini dibutuhkan perancangan basis data, perancangan antarmuka grafis dan perancangan program. Basis data merupakan bagian yang paling inti pada kasus ini, sehingga perlu dirancang dengan struktur data yang tepat supaya mudah untuk mengantisipasi adanya tabrakan jadwal, baik itu jadwal persemester, jadwal dosen maupun jadwal pemakaian ruang kuliah.

Perancangan antarmuka grafis yang baik akan mendukung kemudahan dan kelancaran pemakaian sistem yang dibuat, oleh karena itu harus memenuhi kriteria seperti yang disebutkan pada Tinjauan Pustaka di atas.

Perancangan program akan mengatur urutan pemeriksaan tabrakan jadwal dan pemilihan perintah SQL SELECT yang efektif.

2.11. Perancangan Basis Data

Inti dari tabel yang akan diperiksa adanya tabrakan atau tidak adalah tabel Jadwal, dalam hal ini penulis membuat rancangan basis data seperti Gambar 1 di bawah ini:



Gambar 1 Desain basis data Jadwal Kuliah

Prosiding
ANNUAL RESEARCH SEMINAR 2016
6 Desember 2016, Vol 2 No. 1

Struktur data penting pada basis data ini adalah kolom-kolom yang berada pada **TabelJadwal**, yaitu : hari, jamMulai, jamSelesai, kodeMataKuliah, grup, nik, kodeRuang dan selain itu kolom untukSemester pada **TabelMataKuliah**.

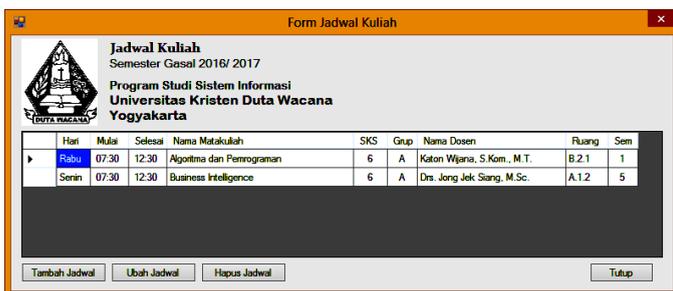
Untuk mendeteksi tabrakan semester dibutuhkan kolom dari **TabelJadwal** : hari, jamMulai, jamSelesai, grup dan kolom untukSemester pada **TabelMataKuliah**

Untuk mendeteksi tabrakan jadwal dosen dibutuhkan kolom dari **TabelJadwal** : hari, jamMulai, jamSelesai dan nik.

Untuk mendeteksi tabrakan ruang dosen dibutuhkan kolom dari **TabelJadwal** : hari, jamMulai, jamSelesai dan kodeRuang.

2.12. Antarmuka Grafis

Antarmuka grafis utama pada penelitian ini adalah FormJadwal, yang sekaligus menampilkan jadwal-jadwal yang sudah disusun sebelumnya. Untuk menambahkan, mengubah maupun menghapus pemakai dapat melakukannya dengan menekan tombol yang sudah disediakan seperti Gambar 2 berikut ini:



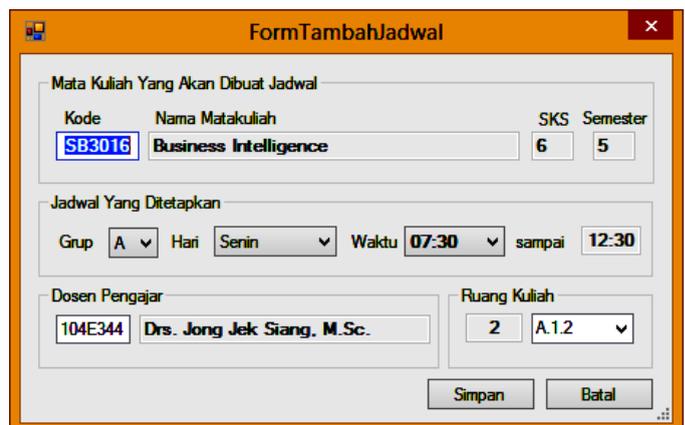
Gambar 2 Antarmuka grafis Form Jadwal Kuliah

Apabila pemakai hendak menambahkan jadwal dengan menekan tombol "Tambah Jadwal" maka program akan beralih ke antarmuka grafis untuk mengisikan jadwal baru seperti Gambar 3 di bawah ini:



Gambar 3 Antarmuka grafis untuk menambahkan Jadwal Kuliah

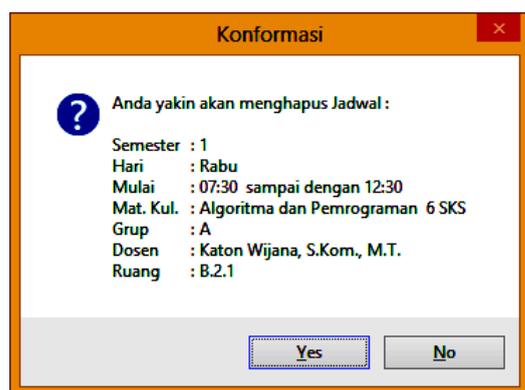
Apabila pemakai hendak mengubah jadwal dengan menekan tombol "Ubah Jadwal" maka antarmuka grafis yang tampil adalah sama seperti Gambar 3 namun jadwal lama yang dipilih akan dituliskan sebagai nilai default antarmuka grafis. Sebagai contoh misalnya jadwal yang dipilih adalah jadwal kedua "Business Intelligence" seperti tampak pada Gambar 2, maka antarmuka grafis yang tampil adalah seperti Gambar 4 di bawah ini:



Gambar 4 Antarmuka grafis untuk mengubah Jadwal Kuliah

Pada kasus ini ketika jadwal sedang diubah maka jadwal tersebut akan dihapus dari TabelJadwal untuk memudahkan proses deteksi tabrakan jadwal baru.

Apabila pemakai hendak menghapus jadwal dengan menekan tombol "Hapus Jadwal" maka sebelum dihapus program akan melakukan konfirmasi melalui kotak pesan seperti Gambar 5 di bawah ini:



Gambar 5 Kotak pesan konfirmasi penghapusan

Prosiding
ANNUAL RESEARCH SEMINAR 2016

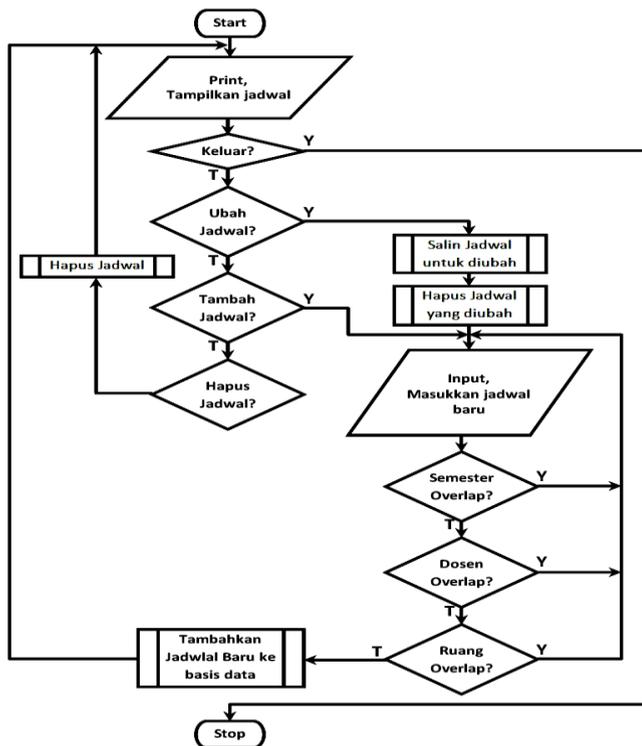
6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI

http://ars.ilkom.unsri.ac.id

2.13. *Perancangan Program*

Program utama pada program bantu penyusunan jadwal ini adalah seperti Gambar 2 Form Jadwal Kuliah, diagram alir untuk program secara keseluruhan adalah seperti Gambar 6 di bawah ini:



Gambar 6 Flow chart Program Bantu Penyusunan Jadwal Kuliah

IV. ANALISIS

Program dibuat memakai bahasa pemrograman C# dengan memanfaatkan .NET Class Library yang melibatkan Windows Form (WinForm) dan ADO.NET.

Menampilkan Jadwal

Untuk menampilkan Jadwal kuliah pada DataGridView seperti pada Gambar 2, diperlukan View sebagai berikut:

```
SELECT dbo.TabelJadwal.kodeJadwal,
dbo.TabelJadwal.hari, dbo.TabelJadwal.jamMulai,
dbo.TabelJadwal.jamSelesai,
dbo.TabelJadwal.kodeMataKuliah,
dbo.TabelMataKuliah.namaMataKuliah,
dbo.TabelMataKuliah.sks, dbo.TabelJadwal.grup,
dbo.TabelJadwal.nik, dbo.TabelDosen.namaDosen,
dbo.TabelJadwal.kodeRuang, dbo.TabelRuang.namaRuang,
dbo.TabelMataKuliah.untukSemester
FROM dbo.TabelJadwal INNER JOIN
dbo.TabelMataKuliah ON
```

```
dbo.TabelJadwal.kodeMataKuliah =
dbo.TabelMataKuliah.kodeMataKuliah INNER JOIN
dbo.TabelDosen ON dbo.TabelJadwal.nik =
dbo.TabelDosen.nik INNER JOIN
dbo.TabelRuang ON dbo.TabelJadwal.kodeRuang =
dbo.TabelRuang.kodeRuang
```

Menambah Jadwal

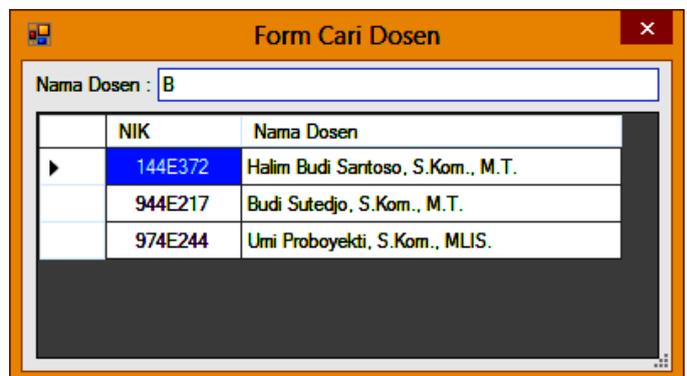
Untuk menambah jadwal baru akan dilakukan penambahan baris pada TabelJadwal dengan nilai-nilai yang dibutuhkan untuk kolom : hari, jamMulai, jamSelesai, kodeMataKuliah, grup, nik dan kodeRuang sedangkan kodeJadwal akan terisi data integer secara otomatis (*auto increment*). Oleh karena itu ketika pemakai menekan tombol "Tambah Jadwal" pada FormJadwalKuliah, maka akan ditampilkan antarmuka grafis FormTambahJadwal seperti tampak pada Gambar 3 yang menyediakan data untuk semua kolom TabelJadwal.

Sesuai dengan petunjuk pada tinjauan pustaka mengenai Interaksi Manusia dan Komputer, untuk pengisian kode mata kuliah disediakan layar bantuan yang dapat menyaring data untuk memilih mata kuliah karena bagi pemakai lebih mudah mengingat nama mata kuliah daripada kodenya, antarmuka grafis untuk penyaringan tersebut adalah seperti Gambar 7 di bawah ini:



Gambar 7 Form bantuan untuk mengisi kode mata kuliah

Sedangkan untuk memasukkan nik dosen, juga disediakan antarmuka grafis bantuan seperti antarmuka grafis pencarian mata kuliah seperti Gambar 8 di bawah ini:



Gambar 8 Form bantuan untuk mengisi nik dosen

Prosiding
ANNUAL RESEARCH SEMINAR 2016

6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI

http://ars.ilkom.unsri.ac.id

Demikian juga perhitungan waktu selesai, karena dapat dilakukan oleh program maka bagian ini tidak perlu diisi oleh pemakai dengan cara melakukan perhitungan dari data jam mulai dan bobot SKS mata kuliah. Pada program ini Jam pada kolom jamMulai dan jamSelesai adalah memakai tipe String dengan format HH:MM, oleh karena itu perlu dibuat metode untuk melakukan konversi dari HH:MM tersebut ke dalam menit sebagai berikut:

```
int toMenit(String waktu)
{
    int jam = int.Parse(waktu.Substring(0, 2));
    int men = int.Parse(waktu.Substring(3));
    return jam * 60 + men;
}
```

Kemudian setelah dilakukan proses perhitungan, yaitu ditambah dengan 50 x SKS (1 SKS = 50 menit), maka untuk ditampilkan kembali perlu dikembalikan menjadi format HH:MM sebagai berikut:

```
String toWaktu(int menit)
{
    String waktu = "";
    String jam = (menit / 60).ToString().PadLeft(2, '0');
    String men = (menit % 60).ToString().PadLeft(2, '0');
    waktu = jam + ":" + men;
    return waktu;
}
```

Metode tersebut dibutuhkan setiap ada perubahan mata kuliah dan jamMulai (ComboBox), oleh karena itu perlu dibuat metode untuk menghitung seperti di bawah ini:

```
void hitungJamSelesai()
{
    int durasi = int.Parse(txtSks.Text) * 50;
    txtJamSelesai.Text =
        toWaktu(toMenit(cbJamMulai.Text) + durasi);
}
```

Setelah FormTambahJadwal terisi dengan lengkap seperti Gambar 4 dan pemakai menekan tombol "Simpan", maka nilai-nilai yang dibutuhkan untuk menambah baris pada TabelJadwal disimpan ke dalam struktur data dalam bentuk kelas yang sudah dipersiapkan seperti berikut:

```
public class Jadwal
{
    public String hari, jamMulai, jamSelesai,
        kodeMataKuliah, grup, nik, kodeRuang;
    public Jadwal()
    {
        hari = "";
        jamMulai = "";
        jamSelesai = "";
    }
}
```

```
kodeMataKuliah = "";
grup = "";
nik = "";
kodeRuang = "";
}
}
```

Tempat penyimpanannya adalah berupa variabel anggota kelas FormTambahJadwal seperti berikut:

```
public Jadwal jadwalBaru = new Jadwal();
```

Maka pada waktu tombol "Simpan" ditekan data penting akan dipindahkan ke variabel anggota tersebut sebagai berikut:

```
jadwalBaru.hari = cbHari.Text;
jadwalBaru.jamMulai = cbJamMulai.Text;
jadwalBaru.jamSelesai = txtJamSelesai.Text;
jadwalBaru.kodeMataKuliah = txtKodeMataKuliah.Text;
jadwalBaru.grup = cbGrup.Text;
jadwalBaru.nik = txtNik.Text;
jadwalBaru.kodeRuang = txtKodeRuang.Text;
```

Kemudian aliran program akan kembali ke tombol "Tambah Jadwal" pada FormJadwalKuliah, di sini akan dilakukan penambahan baris pada TabelJadwal berdasarkan nilai-nilai yang telah tersimpan dalam variabel anggota jadwalBaru tadi. Berikut ini adalah kode program pada tombol "Tambah Jadwal" selengkapnya:

```
private void btnAdd_Click(object sender, EventArgs e)
{
    FormTambahJadwal frmTambahJadwal = new
    FormTambahJadwal();
    frmTambahJadwal.ShowDialog();
    if (frmTambahJadwal.jadwalBaru.kodeMataKuliah.Length
    == 6)
    {
        String strSQL = "INSERT INTO TabelJadwal" +
            "(hari, jamMulai, jamSelesai, kodeMataKuliah," +
            "grup, nik, kodeRuang) VALUES('{0}','{1}'," +
            "'{2}','{3}','{4}','{5}','{6}')";
        Jadwal j = frmTambahJadwal.jadwalBaru;
        strSQL = String.Format(strSQL, j.hari,
            j.jamMulai, j.jamSelesai,
            j.kodeMataKuliah, j.grup, j.nik,
            j.kodeRuang);
        frmTambahJadwal.Dispose();

        //simpan jadwal baru:
        view_jadwal.eksekusiSQL(strSQL);
    }
    else
}
```

Prosiding ANNUAL RESEARCH SEMINAR 2016

6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI

<http://ars.ilkom.unsri.ac.id>

```
    MessageBox.Show("Jadwal baru batal...");  
}
```

Deteksi Tabrakan

Pada saat tombol "Simpan" FormTambahJadwal ditekan, sebelumnya dilakukan pengecekan 3 kemungkinan tabrakan, yaitu tabrakan semester, tabrakan dosen dan tabrakan ruang.

Berikut ini adalah Query yang dapat dipergunakan untuk mendeteksi tabrakan semester :

```
String cekSemester()  
{  
    //Periksa : untukSemester, hari, jamMulai, JamSelesai, grup  
    String strSQL = "SELECT * FROM View_Jadwal WHERE  
    untukSemester='{0}' AND hari='{1}' " +  
    "AND grup='{2}' AND " +  
    "((jamMulai>='{3}' AND jamMulai<='{4}') OR " +  
    "(jamSelesai>='{3}' AND jamSelesai<='{4}')");  
    strSQL = String.Format(strSQL, txtSemester.Text,  
    cbHari.Text, cbGrup.Text,  
    cbJamMulai.Text, txtJamSelesai.Text);  
    String adaJadwal = view_jadwal.adaJadwal(strSQL);  
    return adaJadwal;  
}
```

Berikut ini adalah Query yang dapat dipergunakan untuk mendeteksi tabrakan jadwal dosen :

```
String cekDosen()  
{  
    //Periksa : nik, hari, jamMulai, JamSelesai  
    String strSQL = "SELECT * FROM View_Jadwal WHERE  
    nik='{0}' AND hari='{1}' AND " +  
    "((jamMulai>='{2}' AND jamMulai<='{3}') OR " +  
    "(jamSelesai>='{2}' AND jamSelesai<='{3}')");  
    strSQL = String.Format(strSQL, txtNik.Text,  
    cbHari.Text,  
    cbJamMulai.Text, txtJamSelesai.Text);  
    String adaJadwal = view_jadwal.adaJadwal(strSQL);  
    return adaJadwal;  
}
```

Berikut ini adalah Query yang dapat dipergunakan untuk mendeteksi tabrakan jadwal pemakaian ruang kuliah :

```
String cekRuang()  
{  
    //Periksa : kodeRuang, hari, jamMulai, JamSelesai  
    String strSQL = "SELECT * FROM View_Jadwal WHERE  
    kodeRuang='{0}' AND hari='{1}' AND " +  
    "((jamMulai>='{2}' AND jamMulai<='{3}') OR " +  
    "(jamSelesai>='{2}' AND jamSelesai<='{3}')");  
    strSQL = String.Format(strSQL, txtKodeRuang.Text,  
    cbHari.Text,
```

```
    cbJamMulai.Text, txtJamSelesai.Text);  
    String adaJadwal = view_jadwal.adaJadwal(strSQL);  
    return adaJadwal;  
}
```

Mengubah Jadwal

Untuk mengubah jadwal dilakukan dengan cara data jadwal lama yang hendak diubah disalin terlebih dahulu untuk dikirim ke FormTambahJadwal kemudian jadwal lama tersebut dihapus dari TabelJadwal. Tampilan awal Antarmuka grafis pada Form Tambah Jadwal akan sudah terisi dengan data jadwal yang hendak diubah seperti Gambar 4.

Penghapusan jadwal yang akan diubah ini dilakukan agar pengecekan tabrakan jadwal lebih mudah karena tidak jadwal yang sedang diubah tidak perlu diperhitungkan. Jadi pada program ini perubahan jadwal dilakukan dengan cara menghapus data jadwal terlebih dahulu kemudian menambah data jadwal seperti jadwal baru. Program selengkapnya adalah seperti berikut:

```
String kodeJadwal, kodeMataKuliah, namaMataKuliah, sks,  
semester, grup, hari, jamMulai, jamSelesai, nik,  
namaDosen, namaRuang;  
int baris = dgvJadwal.SelectedCells[0].RowIndex;  
kodeJadwal =  
dgvJadwal.Rows[baris].Cells[0].Value.ToString();  
kodeMataKuliah =  
dgvJadwal.Rows[baris].Cells[4].Value.ToString();  
namaMataKuliah =  
dgvJadwal.Rows[baris].Cells[5].Value.ToString();  
sks = dgvJadwal.Rows[baris].Cells[6].Value.ToString();  
semester =  
dgvJadwal.Rows[baris].Cells[12].Value.ToString();  
grup = dgvJadwal.Rows[baris].Cells[7].Value.ToString();  
hari = dgvJadwal.Rows[baris].Cells[1].Value.ToString();  
jamMulai =  
dgvJadwal.Rows[baris].Cells[2].Value.ToString();  
jamSelesai =  
dgvJadwal.Rows[baris].Cells[3].Value.ToString();  
nik = dgvJadwal.Rows[baris].Cells[8].Value.ToString();  
namaDosen =  
dgvJadwal.Rows[baris].Cells[9].Value.ToString();  
namaRuang =  
dgvJadwal.Rows[baris].Cells[11].Value.ToString();  
view_jadwal.eksekusiSQL("DELETE FROM TabelJadwal WHERE  
kodeJadwal='" + kodeJadwal + "'");  
FormTambahJadwal frmEditJadwal = new  
FormTambahJadwal(kodeMataKuliah, namaMataKuliah, sks,  
semester, grup, hari, jamMulai, jamSelesai, nik,  
namaDosen, namaRuang);  
frmEditJadwal.ShowDialog();  
if (frmEditJadwal.jadwalBaru.kodeMataKuliah.Length == 6)  
{  
    String strSQL = "INSERT INTO TabelJadwal" +  
    "(hari, jamMulai, jamSelesai, kodeMataKuliah,  
    grup, nik, kodeRuang)" +
```

Prosiding
ANNUAL RESEARCH SEMINAR 2016

6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI

http://ars.ilkom.unsri.ac.id

```
"
VALUES('{0}','{1}','{2}','{3}','{4}','{5}','{6}')";
    Jadwal j = frmEditJadwal.jadwalBaru;
    strSQL = String.Format(strSQL, j.hari, j.jamMulai,
j.jamSelesai,
    j.kodeMataKuliah, j.grup, j.nik, j.kodeRuang);
    frmEditJadwal.Dispose();

    //simpan jadwal baru:
    view_jadwal.eksekusiSQL(strSQL);
}
else
    MessageBox.Show("Jadwal Pengganti batal...");
```

Menghapus Jadwal

Karena kunci utama TabelJadwal pada pembahasan ini tidak direlasikan ke tabel lain, sehingga tidak ada konstrain integritas yang perlu diperhatikan, maka penghapusan jadwal dapat dilakukan secara langsung pada TabelJadwal.

Untuk memastikan bahwa pemakai akan menghapus jadwal tersebut, sebelumnya data tiap kolom pada baris DataGridView yang dipilih disalin terlebih dulu ke variabel lokal untuk memudahkan pembuatan kota pesan yang menampilkan data jadwal yang akan dihapus tersebut. Program selengkapnya adalah seperti berikut:

```
String kodeJadwal, namaMataKuliah, sks, semester, grup,
hari, jamMulai, jamSelesai, namaDosen, namaRuang;
int baris = dgvJadwal.SelectedCells[0].RowIndex;
kodeJadwal =
dgvJadwal.Rows[baris].Cells[0].Value.ToString();
namaMataKuliah =
dgvJadwal.Rows[baris].Cells[5].Value.ToString();
sks = dgvJadwal.Rows[baris].Cells[6].Value.ToString();
semester =
dgvJadwal.Rows[baris].Cells[12].Value.ToString();
grup = dgvJadwal.Rows[baris].Cells[7].Value.ToString();
hari = dgvJadwal.Rows[baris].Cells[1].Value.ToString();
jamMulai =
dgvJadwal.Rows[baris].Cells[2].Value.ToString();
jamSelesai =
dgvJadwal.Rows[baris].Cells[3].Value.ToString();
namaDosen =
dgvJadwal.Rows[baris].Cells[9].Value.ToString();
```

```
namaRuang =
dgvJadwal.Rows[baris].Cells[11].Value.ToString();

String hapus = "Anda yakin akan menghapus Jadwal :\n" +
"\nSemester \t: " + semester +
"\nHari \t: " + hari +
"\nMulai \t: " + jamMulai +
" sampai dengan " + jamSelesai +
"\nMat. Kul. \t: " + namaMataKuliah + " " +sks +
" SKS\nGrup \t: " + grup +
" \nDosen \t: " + namaDosen +
"\nRuang \t: " + namaRuang;
```

```
DialogResult jwb = MessageBox.Show(hapus, "Konfirmasi",
MessageBoxButtons.YesNo, MessageBoxIcon.Question);
if (jwb == DialogResult.Yes)
{
    view_jadwal.eksekusiSQL("DELETE FROM TabelJadwal
WHERE kodeJadwal='" + kodeJadwal + "'");
    MessageBox.Show("Penghapusan selesai...");
}
else
{
    MessageBox.Show("Penghapusan dibatalkan...");
}
```

REFERENSI

- [1] Coronel, Morris and Rob, "Database systems design, implementation ana management," Cengage Learning. Boston, pp. 11-13, 2013. (*Referensi*)
- [2] Mark Spenik and Orryn Sledge, "Microsoft SQL Server 2000 DBA Survival Guide," Sams Publishing. Indianapolis, pp. 559-574, 2001. (*Referensi*)
- [3] Kenneth E. Kendall and Julie E. Kendall, "System analysis and design," Prentice Hall. Upper Saddle River, New Jersey, pp. 555-556, 2006. (*Referensi*)
- [4] Raharjo Budi, "Mudah belajar C#," Penerbit Informatika. Bandung, pp. 2, 2015. (*Referensi*)
- [5] H.M. Dietel, P.J. Dietel, J.A. Listfield, T.R. Nieto, C.H. Yaeger and M. Zlatkina, "C#: A Programmer's introduction," Prentice Hall. Upper Saddle River, New Jersey, pp. 361, 2003. (*Referensi*)