



PETIR

JURNAL PENGKAJIAN DAN PENERAPAN TEKNIK INFORMATIKA

VOLUME 9 - NOMOR 1

MARET 2016

ISSN 1978-9262

PENENTUAN NASABAH PENERIMA REWARD PRODUK *GOLD* DENGAN METODE *SIMPLE ADDITIVE WEIGHTING* (SAW)
STUDI KASUS : PT. PINJAM INDONESIA

Riki Ruli A. Siregar; Faizal Fachrurrozi

IMPLEMENTASI METODE *BACKWARD CHAINING* PADA DATA *WAREHOUSE* DAOP 1 JAKARTA PT. KAI (PERSERO)

Rakhmat Arianto; Chandra Bagus Sugiarto

IMPLEMENTASI METODE DETEKSI TEPI *CANNY* PADA OBJEK SEBAGAI MODEL KEAMANAN APLIKASI PADA
SMARTPHONE ANDROID

Abdul Haris; Andi Prasetyo

ANALISA DATA DAN PERANCANGAN APLIKASI SERVICE PELANGGAN PT. JNE UNTUK PENINGKATAN KUALITAS
LAYANAN

Dewi Arianti Wulandari; Sonny Syahrindra Putra

JARINGAN AD-HOC VEHICULAR (VANET) : TINJAUAN TENTANG ARSITEKTUR, KARAKTERISTIK, APLIKASI, DAN
PROTOKOL *MEDIUM ACCESS CONTROL* (MAC)

Rosida Nur Aziza

PERANCANGAN APLIKASI PENJADWALAN MATA KULIAH MENGGUNAKAN METODE *CONSTRAINT PROGRAMMING*

Syam Gunawan

RANCANGAN TATA KELOLA PEREMAJAAN RUANG KELAS DIREKTORAT PEMBINAAN SEKOLAH DASAR

Ratna Mutu Manikam; Purwanto

PENGEMBANGAN AMORIK MENGGUNAKAN METODE GARIS SINGGUNG TERHADAP DUA LINGKARAN DAN PERSAMAAN
KURVA BEZIER ORDE DUA.

Darma Rusjdi

OPTIMALISASI PENGAMBILAN KEPUTUSAN PENILAIAN KINERJA DOSEN PADA PERGURUAN TINGGI ISLAM XYZ
MENGGUNAKAN AHP (ANALYTICAL HIERARCHY PROCESS)

Rahma Farah Ningrum

PENGAMANAN SMS PADA TELEPON SELULER BERBASIS ANDROID MENGGUNAKAN ALGORITMA TRIPLE DES

Raka Yusuf; M. Rival Suheri

APLIKASI LATIHAN SOAL UJIAN TEORI SURAT IZIN MENGEMUDI BERBASIS WEB

Harni Kusniyati; Raka Yusuf; Andri Setiawan

RANCANG BANGUN SIMULASI TERJADINYA LISTRIK DENGAN SUMBER DAYA SAMPAH BERBASIS MULTIMEDIA
(STUDI KASUS : TPST BANTAR GEBANG)

Yasni Djamain; Ika Fitriyani Putri

ISSN 1978-9262



771978 926272

SEKOLAH TINGGI TEKNIK - PLN (STT-PLN)

PETIR

VOL. 9

NO. 1

HAL. 1 - 87

JAKARTA, MARET 2016

ISSN 1978-9262

PERANCANGAN APLIKASI PENJADWALAN MATA KULIAH MENGGUNAKAN METODE *CONSTRAINT PROGRAMMING*

Syam Gunawan

Program Studi Sistem Informasi, STMIK-INDONESIA, Jakarta
syam.gun@gmail.com

ABSTRAK

Perguruan tinggi adalah satuan pendidikan penyelenggara pendidikan tinggi. Ujung tombak kegiatan pembelajaran di Perguruan Tinggi berada di Program Studi. Program studi harus mampu mengatur diri sendiri serta memberikan pelayanan prima. Dalam upaya meningkatkan efisiensi, efektivitas, transparansi dan akuntabilitas penyelenggaraan program studi maka perlu adanya Sistem Informasi Akademik. Salah satu komponen penting dalam kegiatan pembelajaran adalah Jadwal Perkuliahan. Jadwal Perkuliahan adalah daftar yang memuat atau berisi nama mata kuliah, dosen mata kuliah, waktu, ruang perkuliahan dan lain sebagainya. Jadwal Perkuliahan harus sudah tersedia sebelum kegiatan pelaksanaan perkuliahan dilaksanakan. Membangun Jadwal Perkuliahan untuk suatu program studi yang besar memang merupakan suatu pekerjaan yang kompleks, sehingga dalam pelaksanaannya sering terjadi tabrakan jam, ruang, dosen mempunyai jadwal yang sama untuk mata kuliah yang berbeda, mahasiswa mengikuti perkuliahan pada waktu yang sama pada mata kuliah yang berbeda dan lain sebagainya. Hal ini jika dilihat sepintas merupakan masalah yang ringan, namun jika dibiarkan akan sangat berpengaruh pada kegiatan akademik secara keseluruhan. Oleh karena itu, Jadwal Perkuliahan sebaiknya dimasukkan sebagai bagian dari program Sistem Informasi Akademik.

Kata Kunci: *jadwal kuliah, Constraint Programming, Sistem Informasi Akademik.*

PENDAHULUAN

Kegiatan perkuliahan umumnya dilakukan melalui tatap muka antara pengajar (dosen) dengan mahasiswa dalam waktu tertentu yang disusun dalam jadwal mata kuliah. Penjadwalan mata kuliah ini dilakukan setiap pergantian semester dan membutuhkan waktu, tenaga serta ketelitian untuk membuatnya.

Dalam pembuatan jadwal mata kuliah harus memenuhi aturan-aturan penjadwalan untuk menghindari terjadinya masalah seperti adanya jadwal dosen yang mengajar mata kuliah yang berbeda dalam waktu bersamaan atau tidak tersedianya ruangan untuk mata kuliah tertentu.

Untuk mengatasi permasalahan di atas, dibutuhkan suatu sistem pendukung perkuliahan untuk membuat jadwal secara otomatis sehingga dapat meningkatkan efisiensi kerja dari berbagai pihak. Aplikasi ini juga diharapkan dapat membantu mahasiswa yang ingin mengambil mata kuliah lebih dari jumlah SKS paket yang ditawarkan, agar tidak mengalami kesulitan untuk pemilihan mata kuliah yang ingin ditempuh pada semester tersebut.

Dalam Tugas Akhir ini dirancang sebuah aplikasi untuk mengatur jadwal kuliah secara otomatis agar menghasilkan keluaran berupa jadwal kuliah yang optimal, namun jadwal ini masih memungkinkan untuk diubah sesuai dengan keinginan dari user.

Tinjauan Sistem Berjalan

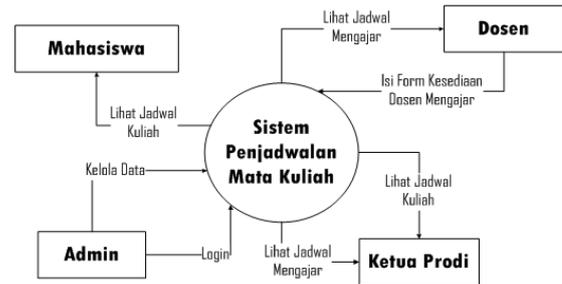
Sistem informasi akademik STMIK Indonesia selama ini sudah menggunakan sistem yang telah terkomputerisasi dan proses pengolahan jadwal

mata kuliah pun sudah dilakukan dengan menggunakan program yang telah terintegrasi dengan menggunakan database, namun hanya bisa diakses di dalam kampus saja atau belum dapat diakses secara *online*. Keadaan demikian mengakibatkan antrian mahasiswa yang akan mengakses jadwal kuliah karena keterbatasan teknologi yang tersedia. Sistem akademik yang telah berjalan memiliki beberapa kelemahan diantaranya pengisian jadwal kuliah bersifat manual yaitu mahasiswa datang ke kampus untuk melihat jadwal program studi dan pelayanan sistem akademik di STMIK Indonesia yang tidak efisien. Salah satunya pada penyampaian informasi jadwal mata kuliah hanya ditampilkan dalam bentuk kertas yang dapat diambil melalui BAAK dan disampaikan melalui papan pengumuman yang dapat menghabiskan waktu dan biaya.



Gambar 1. Flow map Penjadwalan Matakuliah

| | | | |
|--|--|--|---|
| <ul style="list-style-type: none"> Masih terjadi pemborosan dalam penggunaan kertas dalam penyampaian informasi jadwal mata kuliah. | <p>Cause : Hanya ditampilkan dalam bentuk kertas yang dipasang melalui papan pengumuman.</p> <p>Effect : Akan mengakibatkan lambannya proses pengelolaan data jadwal kuliah.</p> | <p>Dengan adanya sistem online ini dapat bermanfaat dalam meminimalkan biaya yang dikeluarkan.</p> | <p>Proses sistem tersebut hanya Pengelolaan jadwal mata kuliah, data dosen, dan data mahasiswa.</p> |
|--|--|--|---|



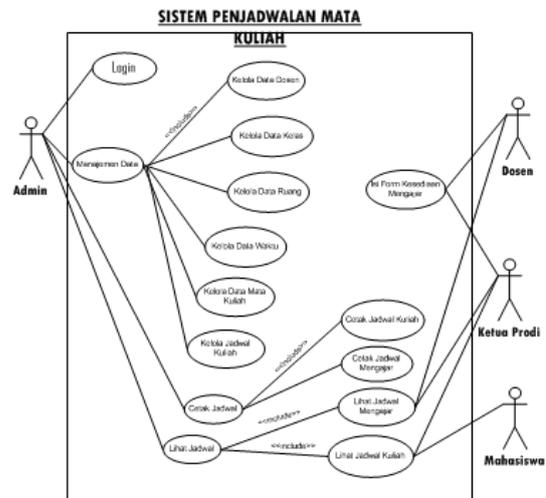
Gambar. Diagram Konteks Usulan

Analisa Masalah

Untuk menyelesaikan suatu masalah, kita harus dapat mengidentifikasi penyebab timbulnya masalah, baru bisa menganalisis sebuah sistem, analisis yang digunakan yaitu analisis pieces. Analisis ini digunakan untuk menganalisa system.

Tabel 3.1. Cause and effect analysis

| Analisa Sebab dan Akibat | | Tujuan-Tujuan Perbaikan Sistem | Perbaikan Sistem |
|---|---|--|--|
| Masalah/pelelang | Cause-Effect Analysis | Tujuan Sistem | Batasan Sistem |
| <ul style="list-style-type: none"> Model Pelayanan dalam penyampaian jadwal kuliah terkesan tidak efisien. | <p>Cause : Sistem yang sedang berjalan hanya bisa diakses di dalam lingkungan kampus saja atau belum online.</p> <p>Effect : Informasi yang disajikan tidak up to date dan sering mengalami keterlambatan</p> | <p>Untuk memudahkan dalam mahasiswa melihat jadwal mata kuliah.</p> | <p>Dari segi service yaitu layanan yang diberikan.</p> |
| <ul style="list-style-type: none"> Informasi yang disajikan tidak up to date. | <p>Cause : Informasi tidak di update dan tidak ada mengontrol secara berkala.</p> <p>Effect : Apabila suatu saat terjadi kesalahan, sangatlah sulit untuk dideteksi.</p> | <p>Dapat mengefisien si waktu bagi mahasiswa dan dosen untuk mengetahui informasi jadwal kuliah yang ada pada STMIK Indonesia.</p> | <p>Sistem ini hanya membahas mengenai informasi yang akan diberikan.</p> |



Gambar. Usecase Diagram

Rancangan Permodelan Constraint Programming

Data-data yang harus dipenuhi dalam Constraint Programming untuk penjadwalan mata kuliah ini adalah sebagai berikut:

- Tiap dosen mendapat tugas mengajar materi-materi tertentu.
- Tiap kelas digunakan untuk tingkat tertentu, dan diadakan pada shift tertentu. Ruang disini berarti "ruang kelas" secara fisik dan juga dimaknai sebagai "kelompok mahasiswa".
- Tiap mata kuliah ditujukan untuk tingkat tertentu.
- Tiap dosen memiliki kesanggupan mengajar tertentu, baik shift, hari, atau jamnya.
- Tiap mata kuliah memiliki jumlah sks.

Dimensi dalam Permodelan *Constraint Programming*

Dimensi adalah himpunan objek yang sama dan terlibat dalam penjadwalan kuliah. Dalam perhitungan, dimensi-dimensi ini menggunakan indeks angka integer untuk menggambarkan isi anggotanya. Jadi, dimensi ini dimodelkan sebagai *array*. Misalnya dimensi D adalah himpunan semua huruf kapital, maka $D = \{A, B, C, D, \dots, Z\}$. Dalam perhitungan *Constraint Programming*, maka D akan menjadi *array* dan dinotasikan sebagai:

- $D[0] = A$
- $D[1] = B$
- $D[2] = C$
- ...
- $D[23] = Z$.

Dimensi-dimensi dasar dalam penjadwalan kuliah ini yaitu:

1. *Shift* (S), adalah *shift-shift* yang ada dalam jadwal kuliah, seperti:
 - a. *Shift* Pagi.
 - b. *Shift* Malam.
 - c. Dan seterusnya.
2. Kelas (K), adalah jumlah total kelas yang digunakan untuk proses belajar-mengajar, seperti:
 - a. Kelas 1.1
 - b. Kelas 2.1
 - c. Dan seterusnya.
3. Jam (J), adalah jam-jam kuliah yang akan diadakan untuk tiap *shift*, seperti:
 - a. Jam 09.00 – 11.00
 - b. Jam 11.00 – 13.00
 - c. Dan seterusnya.
4. Dosen (D), adalah dosen-dosen yang akan mengajar mata kuliah tertentu, seperti:
 - a. Budi
 - b. Dede
 - c. Dan seterusnya.
5. Mata Kuliah (M), adalah mata kuliah yang akan diajarkan, seperti:
 - a. Statistik
 - b. Kalkulus
 - c. Dan seterusnya.
6. Tingkat (T), adalah tingkatan-tingkatan yang ada pada kelas perkuliahan, seperti:
 - a. Tingkat 1
 - b. Tingkat 2
 - c. Dan seterusnya.
7. Hari (H), adalah hari-hari perkuliahan yang ada, seperti:
 - a. Hari Senin
 - b. Hari Selasa
 - c. Dan seterusnya.
8. SKS (S), adalah jumlah sks yang harus dipenuhi setiap mata kuliah yang ada, seperti:
 - a. Statistik memiliki 2 sks.
 - b. Kalkulus memiliki 2 sks.
 - c. Dan seterusnya.

Hubungan Antar Dimensi dalam *Constraint Programming*

Dari masing-masing dimensi dasar di atas, ada hubungan-hubungan khusus dari beberapa dimensi yang berkaitan. Hubungan antar dimensi tersebut yaitu:

1. Dosen-Mata Kuliah (DM), adalah penugasan dosen untuk mengajar mata kuliah tertentu, misalnya:
 - a. Budi mengajar Kalkulus.
 - b. Dan seterusnya.
2. Kelas-*Shift* (KS), adalah penentuan kelas pada *shift* perkuliahan, misalnya:
 - a. Kelas 1.1 untuk *Shift* Pagi.
 - b. Dan seterusnya.
3. Kelas-Tingkat (KT), adalah penetapan kelas sesuai dengan tingkatnya, misalnya:
 - a. Kelas 1.1 adalah tingkat 1.
 - b. Dan seterusnya.
4. Mata Kuliah-Tingkat (MT), adalah penetapan mata kuliah sesuai dengan tingkatnya, misalnya:
 - a. Statistik adalah mata kuliah untuk tingkat 1.
 - b. Dan seterusnya.
5. Dosen-*Shift*-Hari-Jam (DSHJ), adalah data waktu mengajar dosen yang ditentukan jamnya setiap harinya, misalnya:
 - a. Budi mampu mengajar setiap hari dari jam 09.00 sampai jam 14.00
 - b. Dede hanya mampu mengajar hari Kamis saja, dari jam 13.00 sampai jam 15.00
 - c. Dan seterusnya.

Langkah-langkah Algoritma *Constraint Programming*

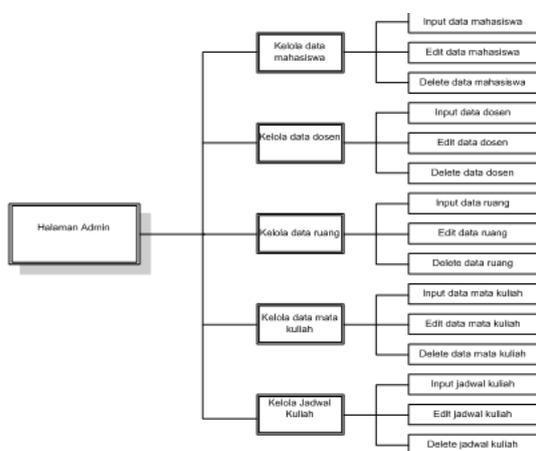
Adapun langkah-langkah algoritma yang dilakukan dalam memasukkan data ke dalam sistem, yaitu:

1. Mendefinisikan data-data dimensi dasar, yaitu:
 - a. *Shift* (S)
 - b. Kelas (K)
 - c. Jam (J)
 - d. Dosen (D)
 - e. Mata Kuliah (M)
 - f. Tingkat (T)
 - g. Hari (H)
 - h. SKS (S).
2. Mendefinisikan hubungan-hubungan antar dimensi yang ada, yaitu:
 - a. Dosen-Mata Kuliah (DM)
 - b. Kelas-*Shift* (KS)
 - c. Kelas-Tingkat (KT)
 - d. Mata Kuliah-Tingkat (MT)
 - e. Dosen-*Shift*-Hari-Jam (DSHJ).
3. Memasukkan variabel solusi ke dalam sistem.
4. Memasukkan semua batasan yang ada ke dalam sistem.
5. Memerintahkan sistem untuk menyeleksi *domain-domain* yang tidak perlu dimasukkan dalam perhitungan solusi.
6. Pencarian solusi, yaitu memerintahkan sistem untuk mencari solusi pertama yang didapat.
7. Pengambilan hasil solusi yang diperoleh dari sistem tersebut.

Tahap-tahap penjadwalan mata kuliah :

1. Pemilihan tahun ajaran dan semester
2. Penentuan ruangan dan *shift*
3. Pemilihan *soft constraint* dan prioritasnya
4. Pemilihan mata kuliah yang tidak membutuhkan ruang kelas

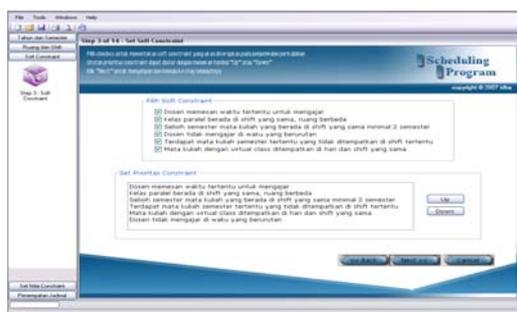
5. Pemilihan slot waktu dan ruang kosong yang tidak diisi kegiatan perkuliahan.
6. Pemilihan mata kuliah yang mempunyai jadwal khusus.
7. Pemilihan mata kuliah yang merupakan kelas virtual.
8. Pemilihan dosen yang memesan waktu beserta shift dan hari yang dipesan.
9. Penentuan prioritas dosen yang memesan waktu mengajar.
10. Pemilihan dosen yang tidak menginginkan waktu ajar yang berurutan.
11. Pemilihan waktu (*shift* dan hari) yang tidak boleh ditempati mata kuliah semester tertentu
12. Penempatan jadwal mata kuliah
13. Pemilihan mata kuliah yang tidak mengadakan ujian.
14. Penempatan jadwal ujian.



Gambar. Struktur Halaman Admin

Menentukan *Soft Constraint* Beserta Prioritasnya

Tampilan menunjukkan bahwa *item soft constraint* yang dipilih akan ditambahkan pada *list box*. Sedangkan *item* yang tidak dipilih akan dihilangkan dari *listbox*. Setelah pilihan *soft constraint* muncul pada *listbox*, maka urutan prioritas batasan tersebut dapat diatur melalui tombol *Up* atau tombol *Down*. *Item constraint* yang dipilih akan naik 1 tingkatan dalam prioritas saat pengguna menekan tombol *Up*. Sebaliknya, jika pengguna menekan tombol *Down*, *item constraint* yang dipilih akan turun 1 tingkat prioritas.



Gambar. Tampilan Constrains

Kesimpulan dan Saran

Dengan menggunakan *constraint programming* maka

1. Membantu bagian administrasi Program Studi Sistem informasi dalam mengontrol data jadwal perkuliahan dalam penyajian informasi jadwal kuliah sehingga dapat dilakukan dengan cepat.
2. Memberikan informasi penjadwalan mata kuliah sesuai dengan kebutuhan.

Saran

Banyak sekali metode yang bias digunakan untuk membuat penjadwalan matakuliah, penulis menyarankan bagi yang ingin membuat tema yang sama bias menggunakan metode yang lain.

DAFTAR PUSTAKA

Nugroho, Eko. *Sistem Informasi Manajemen Konsep, Aplikasi, dan Perkembangan*. Yogyakarta: Andi, 2008.

Teguh Wahyono, 2004, *Sistem Informasi : Konsep Dasar, Analisis Desain dan Implementasi*, Graha Ilmu, Yogyakarta.

Wikipedia, 2014. *Constraint Programming*. (Online). (http://en.wikipedia.org/wiki/Constraint_programming, diakses 25 Juni 2014).

Lestari, Dinda, 2010. *Model-Model Pengembangan Sistem Informasi Berbasis Web, Fakultas Ilmu Kompute, Universitas Sriwijaya*.

Arista, Gea, 2013. *Logical Record Structure (LRS)*. (<http://gea-gas.blogspot.com/2013/04/logical-record-structured-lrs.html>, diakses 10 Juli 2014).

Yulia, Fajar, 2008. *Penjadwalan Kereta Api Tunggal*. (<http://digilib.itb.ac.id/files/disk1/625/jbptitbpgdl-fajaryulia-31250-2-2008ta-1.pdf.html>, diakses 4 Juli 2014).

Anonymous, 2007. *Landasan Teori Constraint Satisfaction Problem*. (<http://thesis.binus.ac.id/Asli/Bab2/2007-2-00451-MTIF%20Bab2.pdf>, diakses 4 Juli 2014).