

PEMODELAN INFORMASI DENGAN METODE FCO-IM

Rina Sibuea

Program Studi Manajemen Informatika, Politeknik Informatika Del
Jl. Sisingamangaraja, Sitoluama, Laguboti, Tobasa, 22381
e-mail : rina@del.ac.id

Abstrak

Pemodelan informasi merupakan aktivitas yang dilakukan untuk membuat suatu model konseptual yang meliputi semua informasi yang relevan dalam bisnis proses. Tujuan pemodelan informasi adalah untuk menyediakan suatu standard untuk proses perkembangan perangkat lunak menggunakan sebuah metode pemodelan, dengan meyakinkan bahwa isi dari nilai standard tersebut benar adanya, konsisten, dan jelas dapat didefenisikan. Salah satu metode pemodelan informasi adalah Fully Communication Information Modeling (FCO-IM). FCO-IM merupakan metode pemodelan informasi konseptual yang berdasarkan bahasa alami. FCO-IM tidak memodelkan struktur dari alam semesta tetapi memodelkan komunikasi tentang alam semesta. Tulisan ini bertujuan untuk menganalisa, mengimplementasikan, dan memodelkan kebutuhan informasi dengan menggunakan metode FCO-IM. Dalam penelitian ini, hasil pemodelan diimplementasikan ke suatu DBMS dan dikembangkan menjadi suatu aplikasi informasi.

Kata kunci : pemodelan informasi, model konseptual, FCO-IM

1. PENDAHULUAN

Informasi memainkan peranan yang sangat penting saat ini.. Informasi merupakan dasar/awal dari pembangunan sistem informasi. Sistem informasi merupakan sebuah sub-sistem dari sebuah organisasi dengan tujuan untuk mendukung kinerja organisasi tersebut sebaik mungkin.

Dalam sistem informasi terdapat *user* yang merekam informasi, memelihara informasi, dan menggambarkan informasi. Sehingga sistem yang hendak dibangun sebaiknya mencakup seluruh informasi yang diperlukan organisasi tersebut. Pemodelan informasi menempati pusat dan posisi yang paling penting dan paling banyak dalam perkembangan sistem informasi dengan jantung permasalahannya adalah menggambarkan sebuah model informasi alirannya untuk diketahui aliran informasinya secara pasti.

Salah satu metode pemodelan informasi adalah Fully Communication Oriented – Information Modeling (FCO-IM). FCO-IM memodelkan informasi dengan memakai bahasa alami *user* [1]. Pada penelitian ini, metode FCO-IM akan digunakan untuk memodelkan informasi (skenario) yang terdapat pada IT Telkom, analisis terhadap informasi (skenario) hingga didapatkan DDL (*Data Definition Language*) *script* dari informasi yang dimodelkan, untuk selanjutnya diimplementasikan ke suatu DBMS (*Database Management System*) dan dikembangkan menjadi suatu aplikasi informasi yang bermanfaat.

2. TINJAUAN PUSTAKA

2.1 Pemodelan Informasi

Pemodelan informasi merupakan suatu cara yang dilakukan untuk mendesain aktivitas dari aliran informasi pada suatu organisasi. Dapat juga diartikan untuk memberikan sebuah spesifikasi menyeluruh dari informasi yang menyusun aliran itu. Berikut merupakan tabel yang menggambarkan fase dalam perkembangan sistem informasi dan hasil (proses penting) yang terdapat pada tiap fase. [1]

Tabel 2-1 Fase perkembangan sistem informasi dan hasil tiap fase

Fase	Hasil (proses penting)
Fase Perencanaan	Perencanaan proyek
Fase Defenisi	Fungsi model umum
Fase Analisis	Model konseptual
Fase Desain	Logik dan laporan teknis dokumen
Fase Pembangunan dan Pengujian	Realisasi sistem informasi
Fase Pengenalan	Manual, rencana pengenalan
Fase Pemeliharaan	Dokumentasi tambahan

Proses pemodelan informasi berada pada fase analisis dengan menggambarkan model konseptual sebagai langkah awal dari suatu perkembangan perangkat lunak yang dilakukan secara sistematis tentang suatu sistem yang akan dimodelkan dan dibangun.

2.2 FCO-IM

FCO-IM merupakan metode pemodelan informasi berdasarkan bahasa alami *user*. Adapun kelebihan FCO-IM dibandingkan dengan metode pemodelan informasi lain adalah sebagai berikut [1] [2]:

1. intensitas keikutsertaan *user* tinggi, sehingga menuju ke arah validasi yang baik
2. pendekatan model menghasilkan desain yang lebih baik
3. dokumentasi terintegrasi penuh sehingga mencegah penambahan biaya pemeliharaan
4. meningkatkan dukungan terhadap perkembangan sistem ke arah yang lebih baik
5. mengurangi *time to market* untuk pemodelan/desain, dan realisasi

Dalam melakukan pemodelan terhadap informasi (skenario), FCO-IM melakukannya dalam beberapa tahapan. Berikut langkah-langkah pemodelan informasi dengan metode FCO-IM [1] [2]:

1. *Verbalization*, yaitu user menginputkan kalimat berdasarkan sebuah fakta (*fact expression*) [5]
2. *Clasify dan Qualify*, ialah proses mengelompokkan kalimat ke dalam kelas-kelas dan memberikan nama terhadap masing-masing kelas untuk selanjutnya akan tersimpan di dalam *Repository* dan berupa *fact type (role), object type, dan label type*. [3]
3. *Information Grammar Diagram (IGD)* untuk mendapatkan model informasi berdasarkan kalimat inputan *user*. Model informasi yang dihasilkan sesuai dengan item-item yang terdapat pada *Repository* [3] [5]
4. *Constraint*, yaitu dengan melakukan penambahan *constraint* yang dilakukan pada IGD. Terdapat beberapa *constraint* pada FCO-IM, yaitu sebagai berikut :
 - a. *Value constraint*, untuk membuat batasan pada *label type*
 - b. *Uniqueness constraint* merupakan sebuah *constraint* yang mengindikasikan bahwa nilai dari setiap populasi hanya dapat muncul sekali
 - c. *Totality constraint* ditempatkan pada *role* yang dimainkan oleh *object type*, dimana populasi dari *object type* harus muncul pada *role* tersebut
 - d. *Subset constraint* diberikan ketika populasi dari satu *role* harus menjadi bagian/subset dari populasi *role* lain
 - e. *Equality constraint* merupakan *constraint* yang terdiri dari 2 subset *constraint*
 - f. *Exclusion constraint* diberikan ketika populasi dari *role* tidak dapat memiliki label
 - g. *Cardinality constraint* menentukan bahwa populasi dari *role* muncul dengan nilai tertentu

Pada umumnya, tidak ada metode *systematic* untuk menentukan *subset, equality, exclusion, maupun cardinality constraint*. *Constraint* tersebut dapat muncul dari informasi (skenario) atau dari *interview* dengan *user*.

5. *Grouping, Lexicalizing, dan Reducing (GLR)*, yaitu mengkombinasikan sebanyak mungkin *fact type* ke dalam tabel yang sama tanpa adanya redundansi, mentransformasi *fact type* sehingga setiap *role* memiliki oleh *label type*, dan menghapus *fact type* tertentu. [4] [5]

3. METODE PENELITIAN

Alat bantu yang digunakan untuk memodelkan informasi dengan metode FCO-IM dalam penelitian ini adalah CaseTalk. Bahan/materi yang digunakan antara lain : (1) skenario, yang berisi informasi umum yang relevan dengan informasi dari *user* ; (2) (sampel) data, yang berisi data yang diasumsikan/didapatkan.

Pada awalnya akan terdapat informasi yang diperoleh dari *user* yang disebut sebagai *starting document* (skenario) . Dalam penelitian ini akan ditampilkan 2 *starting document* (skenario) yang dapat dilihat pada Gambar 3-1 dan Gambar 3-2 :

Setiap orang yang memenuhi syarat dan sudah melakukan prosedur-prosedur untuk menjadi mahasiswa/i di IT Telkom akan mendapatkan NIM sebagai pengenalan dirinya. Dengan adanya NIM dapat diketahui nama, tempat tinggal, kelahiran, gender. Siswa tersebut akan disebut alumni ketika sudah lulus (berarti memiliki waktu kelulusan) dan sudah menyelesaikan semua masalah akademik (mata pelajaran, berarti memiliki IPK). Alumni pada umumnya akan mencari pekerjaan (berarti akan diketahui tempat dan lokasi kerja)
--

Gambar 3-1 Skenario 1

1. Dari skenario 1 di atas, penambahan siswa dari seluruh departemen setiap tahunnya, kebanyakan program studi mana, dan periode berapa tahun (S1 atau D3)
2. Ingin dilakukan pencatatan persebaran pekerjaan alumni. Kebanyakan pada kategori perusahaan mana alumni IT Telkom bekerja.

Gambar 3-2 Skenario 2

Dari kedua skenario tersebut dapat disimpulkan sebagai berikut :

- a. Pada skenario 1 didapatkan informasi bahwa seseorang yang memenuhi syarat untuk menjadi mahasiswa IT Telkom akan mendapat NIM, sebagai penanda bahwa dia sah menjadi mahasiswa IT Telkom. Selanjutnya akan dilakukan pencatatan mengenai nama, tempat tinggal, kelahiran, gender, waktu kelulusan dan IPK.
- b. Pada skenario 2, terdapat penambahan informasi, yaitu program studi dari setiap mahasiswa dan persebaran pekerjaan (kategori perusahaan) mereka jika sudah menyelesaikan perkuliahan. Selain skenario , maka (sample) data juga didapatkan *user*. Berikut sample data untuk skenario yang ada :

Tabel 3-1 Mahasiswa

Nama	Tempat Tinggal	Kelahiran	Gender	NIM	Program Studi
Rina Sibuea	Jl Ciawi No 5 Bandung	28 Agustus 1972	Perempuan	113910375	S1 Teknik Informatika
Reysel Christian	Jl Cempaka No 17 Jakarta	7 Juni 1986	Laki-laki	113040357	D3 Teknik Informatika

Tabel 3-2 Alumni

NIM	Kelulusan	Nilai akhir	Pekerjaan
113910375	5 Mei 1996	3.25	Pegawai BCA Jakarta
113040357	28 Maret 2009	3.05	

Setelah didapatkan (sample) data, maka akan dilakukan analisis terhadap informasi yang diperoleh. Pada pemodelan informasi dengan metode FCO-IM, akan dilakukan verbalisasi terlebih dahulu.

Dari skenario 1 tersebut akan dilakukan verbalisasi (*fact expression*) sekaligus kualifikasi dan klasifikasi terhadap *fact expression* tersebut.

Verbalisasi terhadap skenario 1 dapat dilihat pada Gambar 3-3 :

FACT TYPE LEVEL

MAHASISWA IT TELKOM

F1 : Mahasiswa 11391375

F1 : Mahasiswa 113040357

NAME

F2 : 11391375 bernama Rina Sibuea

F2 : 113040357 bernama Reysel Christian

TEMPAT TINGGAL

F3 : 11391375 tinggal di Jl Ciawi No 5 Bandung

F3 : 113040357 tinggal di Jl Cempaka No 27 Jakarta

BIRTHDAY

F4 : 11391375 lahir pada 28 Agustus 1972

F4 : 113040357 lahir pada 7 Juni 1986

GENDER NAME

F5 : 11391375 merupakan perempuan

F5 : 113040357 merupakan laki-laki

DEPARTEMEN

F6 : 11391375 mengambil kuliah Teknik Informatika periode 4 tahun

F6 : 113040357 mengambil kuliah Teknik Informatika periode 3 tahun

LULUS

F7 : 11391375 lulus pada 5 Mei 1996 dengan nilai akhir 3.25

F7 : 113040357 lulus pada 28 Maret 2009 dengan nilai akhir 3.05

ALUMNI

F8 : 11391375 lulus pada 28 5 Mei 1996 dengan nilai akhir 3.25

F8 : 113040357 lulus pada 28 Maret 2009 dengan nilai akhir 3.05

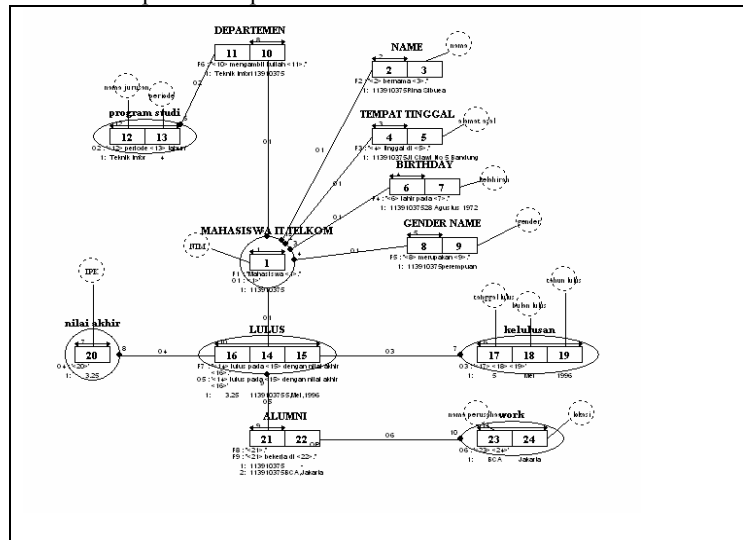
F9 : 11391375 lulus pada 5 Mei 1996 dengan nilai akhir 3.25 bekerja di BCA Jakarta

OBJECT TYPE LEVEL, LABEL TYPE LEVEL

O1 '<NIM>'
 O2 '<nama jurusan> periode <periode> tahun'
 O3 '<tanggal lulus> <bulan lulus> <tahun lulus>'
 O4 '<IPK>'
 O5 '<MAHASISWA IT TELKOM:01> lupus pada <kelulusan:03> dengan nilai akhir <nilai akhir:04>'
 O6 '<nama perusahaan> <lokasi>'
 F1 "Mahasiswa <NIM>"
 F2 "<MAHASISWA IT TELKOM : 01> bernama <nama>"
 F3 "<MAHASISWA IT TELKOM : 01> tinggal di <alamat asal>"
 F4 "<MAHASISWA IT TELKOM : 01> lahir pada <kelahiran>"
 F5 "<MAHASISWA IT TELKOM : 01> merupakan <gender>"
 F6 "<MAHASISWA IT TELKOM : 01> mengambil kuliah <program studi:02>"
 F7 "<MAHASISWA IT TELKOM : 01> lulus pada <kelulusan:03> dengan nilai akhir <nilai akhir:04>"
 F8 "<LULUS:05>"
 F9 "<LULUS:05> bekerja di <work:06>"

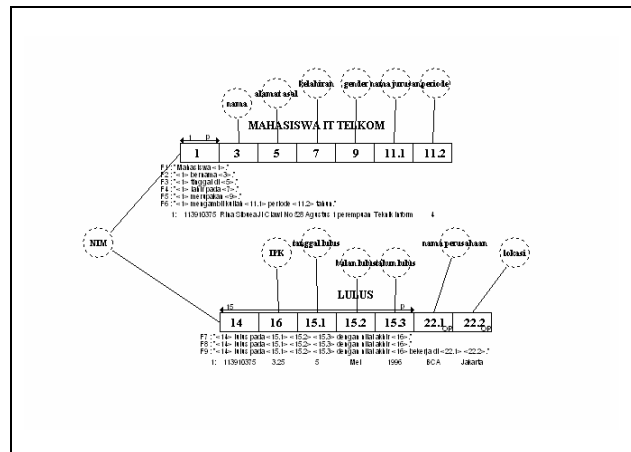
Gambar 3-3 Verbalisasi Skenario 1

Setelah dilakukan verbalisasi, klasifikasi, dan kualifikasi, maka akan diberikan *constraint* sesuai dengan informasi yang diterima dari *user*. Model awal yang dihasilkan dengan metode FCO-IM berdasarkan informasi yang terdapat pada skenario 1 dapat dilihat pada Gambar 3-4 :



Gambar 3-4 : Model FCO-IM Skenario 1

Setelah didapatkan model awal, maka selanjutnya akan dilakukan proses GLR. Proses ini bertujuan untuk menghasilkan sebuah skema *database* yang merekam informasi yang dimodelkan tanpa adanya redundansi. Model akhir dari proses pemodelan dengan metode FCO-IM pada skenario 1 dapat dilihat pada Gambar 3-5:



Gambar 3-5 : Model akhir FCO-IM Skenario 1

Dari model yang didapatkan, akan dihasilkan DDL *script*. Pada penelitian ini, DDL *script* yang dihasilkan merupakan DDL *script* yang di-generate ke dalam relasional database My SQL InnoDB. DDL *script* yang dihasilkan pemodelan informasi (skenario 1) dengan metode FCO-IM dapat dilihat pada Gambar 3-6 .

```

/*****
 * MySQL InnoDB Script generated by CaseTalk. * SQL-DDL Code based on MySQL InnoDB
 * Author : rina/rinds_us45@yahoo.com
 */
/* CREATE DATABASE IF NOT EXISTS skenario 1; */
/***** DROP TABLES *****/
DROP TABLE IF EXISTS `LULUS`;
DROP TABLE IF EXISTS `MAHASISWA_IT_TELKOM`;
/***** CREATE TABLES *****/
/* Table `LULUS`
 *
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>."
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>."
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>
 bekerja di <nama perusahaan> <lokasi>."
 */
CREATE TABLE `LULUS` (
  `MAHASISWA_IT_TELKOM` INTEGER NOT NULL,
  `TANGGAL_LULUS` INTEGER NOT NULL,
  `BULAN_LULUS` CHAR(3) NOT NULL,
  `TAHUN_LULUS` INTEGER NOT NULL,
  `NILAI_AKHIR` NUMERIC NOT NULL,
  `NAMA_PERUSAHAAN` CHAR(3),
  `LOKASI` CHAR(7),
  PRIMARY KEY (`MAHASISWA_IT_TELKOM`, `TANGGAL_LULUS`, `BULAN_LULUS`, `TAHUN_LULUS`,
`NILAI_AKHIR`)
) TYPE=InnoDB;

/* Table `MAHASISWA_IT_TELKOM`
 *
 * "Mahasiswa <NIM>."
 * "<NIM> bernama <nama>."
 * "<NIM> tinggal di <alamat asal>."
 * "<NIM> lahir pada <kelahiran>."
 * "<NIM> merupakan <gender>."
 * "<NIM> mengambil kuliah <nama jurusan> periode <periode> tahun."
 */
CREATE TABLE `MAHASISWA_IT_TELKOM` (
  `NIM` INTEGER NOT NULL,
  `NAMA` CHAR(11) NOT NULL,
  `ALAMAT_ASAL` CHAR(21) NOT NULL,
  `KELAHIRAN` CHAR(15) NOT NULL,
  `GENDER` CHAR(9) NOT NULL,
  `NAMA_JURUSAN` CHAR(18) NOT NULL,
  `PERIODE` INTEGER NOT NULL,
  PRIMARY KEY (`NIM`)
) TYPE=InnoDB;
/***** FOREIGN KEY DEFINITIONS *****/
/* FOREIGN KEYS FOR `LULUS` */
ALTER TABLE `LULUS`
  ADD INDEX `FK1_LULUS` (`MAHASISWA_IT_TELKOM`),
  ADD FOREIGN KEY (`MAHASISWA_IT_TELKOM`)
    REFERENCES `MAHASISWA_IT_TELKOM` (`NIM`);

```

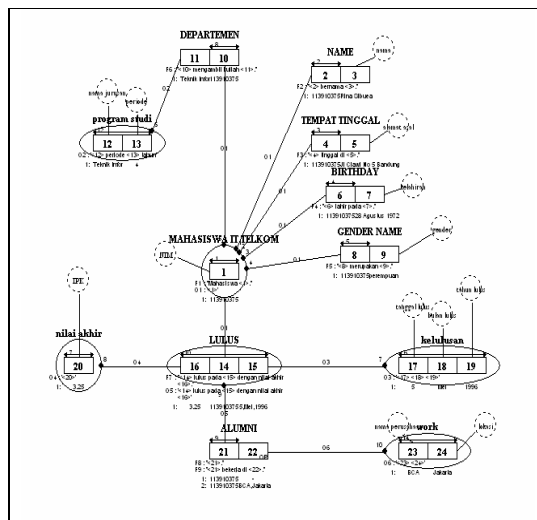
Gambar 3-6 DDL *script* skenario 1

Setelah pemodelan informasi untuk skenario 1 dengan metode FCO-IM, maka akan dilakukan hal yang sama terhadap skenario 2 Verbalisasi, klasifikasi, dan kualifikasi untuk skenario 2 dapat dilihat pada Gambar 3-7:

<p>FACT TYPE LEVEL MAHASISWA IT TELKOM F1 : Mahasiswa 11391375 F1 : Mahasiswa 113040357 NAME F2 : 11391375 bernama Rina Sibuea F2 : 113040357 bernama Reysel Christian TEMPAT TINGGAL F3 : 11391375 tinggal di Jl Ciawi No 5 Bandung F3 : 113040357 tinggal di Jl Cempaka No 27 Jakarta BIRTHDAY F4 : 11391375 lahir pada 28 Agustus 1972 F4 : 113040357 lahir pada 7 Juni 1986 GENDER NAME F5 : 11391375 merupakan perempuan F5 : 113040357 merupakan laki-laki DEPARTEMEN F6 : 11391375 mengambil kuliah Teknik Informatika periode 4 tahun masuk pada tahun 1991 F6 : 113040357 mengambil kuliah Teknik Informatika periode 3 tahun masuk pada tahun 2004 LULUS F7 : 11391375 lulus pada 5 Mei 1996 dengan nilai akhir 3.25 F7 : 113040357 lulus pada 28 Maret 2009 dengan nilai akhir 3.05 ALUMNI F8 : 11391375 lulus pada 5 Mei 1996 dengan nilai akhir 3.25 F8 : 113040357 lulus pada 28 Maret 2009 dengan nilai akhir 3.05 F9 : 11391375 lulus pada 5 Mei 1996 dengan nilai akhir 3.25 bekerja di BCA Jakarta PERUSAHAAN F10 : BCA Jakarta</p>
<p>OBJECT TYPE LEVEL , LABEL TYPE LEVEL O1 '<NIM>' O2 '<nama jurusan> periode <periode> tahun' O3 '<tahun masuk>' O4 '<tanggal lulus> <bulan lulus> <tahun lulus>' O5 '<IPK>' O6 '<MAHASISWA IT TELKOM:01> lupus pada <kelulusan:03> dengan nilai akhir <nilai akhir:05>' O7 '<nama perusahaan> <lokasi>' F1 "Mahasiswa <NIM>" F2 "<MAHASISWA IT TELKOM : 01> bernama <nama>" F3 "<MAHASISWA IT TELKOM : 01> tinggal di <alamat asal>" F4 "<MAHASISWA IT TELKOM : 01> lahir pada <kelahiran>" F5 "<MAHASISWA IT TELKOM : 01> merupakan <gender>" F6 "<MAHASISWA IT TELKOM : 01> mengambil kuliah <program studi:02> masuk pada tahun <angkatan:03>" F7 "<MAHASISWA IT TELKOM : 01> lulus pada <kelulusan:04> dengan nilai akhir <nilai akhir:05>" F8 "<LULUS:06>" F9 "<LULUS:06> bekerja di <work:07>" F 10 "<work:07> termasuk <kategori>"</p>

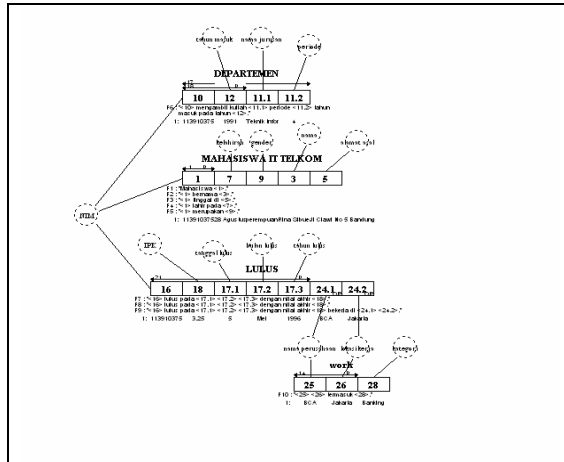
Gambar 3-7 Verbalisasi skenario 2

Model awal yang dihasilkan (setelah penambahan *constraint*) pada skenario 2 dapat dilihat pada Gambar 3-8:



Gambar 3-8: Model FCO-IM Skenario 2

Model akhir (setelah dilakukan proses GLR) pada skenario 2 dapat dilihat pada Gambar 3-9 :



Gambar 3-9 : Model FCO-IM Skenario 2

DDL script yang dihasilkan pada skenario 2 dapat dilihat pada Gambar 3-10 :

```

/***** * MySQL InnoDB Script generated by CaseTalk
 * SQL-DDL Code based on MySQL InnoDB
 * Author : rina/rinds_us45@yahoo.com
 /* CREATE DATABASE IF NOT EXISTS skenario 2; */
/***** DROP TABLES *****/
DROP TABLE IF EXISTS `DEPARTEMEN`;
DROP TABLE IF EXISTS `LULUS`;
DROP TABLE IF EXISTS `MAHASISWA_IT_TELKOM`;
DROP TABLE IF EXISTS `WORK`;
/***** CREATE TABLES *****/
/* Table `DEPARTEMEN`
 *
 * "<NIM> mengambil kuliah <nama jurusan> periode <periode> tahun masuk pada tahun <tahun
 masuk>."
 */
CREATE TABLE `DEPARTEMEN` (
  `MAHASISWA_IT_TELKOM` INTEGER NOT NULL,
  `ANGKATAN` INTEGER NOT NULL,
  `NAMA_JURUSAN` CHAR(18) NOT NULL,
  `PERIODE` INTEGER NOT NULL,
  PRIMARY KEY (`MAHASISWA_IT_TELKOM`, `ANGKATAN`),
  KEY KEY2 (`MAHASISWA_IT_TELKOM`, `NAMA_JURUSAN`, `PERIODE`)
) TYPE=InnoDB;
/* Table `LULUS`
 *
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>."
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>."
 * "<NIM> lulus pada <tanggal lulus> <bulan lulus> <tahun lulus> dengan nilai akhir <IPK>
 bekerja di <nama perusahaan> <lokasi kerja>."
 */
CREATE TABLE `LULUS` (
  `MAHASISWA_IT_TELKOM` INTEGER NOT NULL,
  `TANGGAL_LULUS` INTEGER NOT NULL,
  `BULAN_LULUS` CHAR(3) NOT NULL,
  `TAHUN_LULUS` INTEGER NOT NULL,
  `NILAI_AKHIR` NUMERIC NOT NULL,
  `NAMA_PERUSAHAAN` CHAR(3),
  `LOKASI_KERJA` CHAR(7),
  PRIMARY KEY (`MAHASISWA_IT_TELKOM`, `TANGGAL_LULUS`, `BULAN_LULUS`, `TAHUN_LULUS`,
 `NILAI_AKHIR`)
) TYPE=InnoDB;
/* Table `MAHASISWA_IT_TELKOM`
 *
 * "Mahasiswa <NIM>."
 * "<NIM> bernama <nama>."
 * "<NIM> tinggal di <alamat asal>."
 * "<NIM> lahir pada <kelahiran>."
 * "<NIM> merupakan <gender>."

```

```
*/
CREATE TABLE `MAHASISWA_IT_TELKOM` (
  `NIM` INTEGER NOT NULL,
  `NAMA` CHAR(11) NOT NULL,
  `ALAMAT_ASAL` CHAR(21) NOT NULL,
  `KELAHIRAN` CHAR(15) NOT NULL,
  `GENDER` CHAR(9) NOT NULL,
  PRIMARY KEY (`NIM`)
) TYPE=InnoDB;
/* Table `WORK`
*
* "<nama perusahaan> <lokasi kerja> termasuk <kategori>."
*/
CREATE TABLE `WORK` (
  `NAMA_PERUSAHAAN` CHAR(3) NOT NULL,
  `LOKASI_KERJA` CHAR(7) NOT NULL,
  `KATEGORI` CHAR(7) NOT NULL,
  PRIMARY KEY (`NAMA_PERUSAHAAN`, `LOKASI_KERJA`)
) TYPE=InnoDB;
/***** FOREIGN KEY DEFINITIONS *****/
/* FOREIGN KEYS FOR `DEPARTEMEN` */
ALTER TABLE `DEPARTEMEN`
  ADD INDEX `FK1_DEPARTEMEN` (`MAHASISWA_IT_TELKOM`),
  ADD FOREIGN KEY (`MAHASISWA_IT_TELKOM`)
    REFERENCES `MAHASISWA_IT_TELKOM` (`NIM`);
/* FOREIGN KEYS FOR `LULUS` */
ALTER TABLE `LULUS`
  ADD INDEX `FK1_LULUS` (`MAHASISWA_IT_TELKOM`),
  ADD FOREIGN KEY (`MAHASISWA_IT_TELKOM`)
    REFERENCES `MAHASISWA_IT_TELKOM` (`NIM`);
ALTER TABLE `LULUS`
  ADD INDEX `FK2_LULUS` (`NAMA_PERUSAHAAN`, `LOKASI_KERJA`),
  ADD FOREIGN KEY (`NAMA_PERUSAHAAN`, `LOKASI_KERJA`)
    REFERENCES `WORK` (`NAMA_PERUSAHAAN`, `LOKASI_KERJA`);
/* FOREIGN KEYS FOR `MAHASISWA_IT_TELKOM` */
/* NOT SUPPORTED BY MYSQL:
ALTER TABLE `MAHASISWA_IT_TELKOM`
  ADD CHECK (`NIM`
    IN (SELECT `MAHASISWA_IT_TELKOM` FROM `DEPARTEMEN`));
*/
/* FOREIGN KEYS FOR `WORK` */
/* NOT SUPPORTED BY MYSQL:
ALTER TABLE `WORK`
  ADD CHECK (EXISTS
    (SELECT 1
     FROM `LULUS`
     WHERE `NAMA_PERUSAHAAN` = `WORK`.`NAMA_PERUSAHAAN`
       AND `LOKASI_KERJA` = `WORK`.`LOKASI_KERJA`));
*/
```

Gambar 3-10 DDL script skenario 2

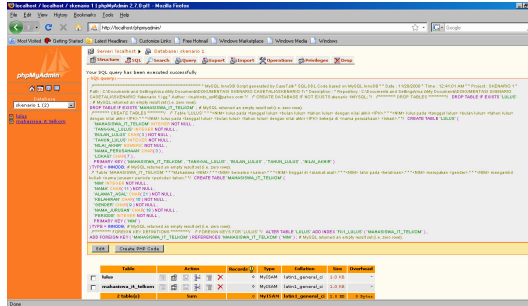
Untuk selanjutnya, skenario 1 dan skenario 2 akan disebut sebagai basisdata (jika DDL script sudah diimplementasikan ke dalam suatu DBMS)

Analisis Hasil Pemodelan FCO-IM :

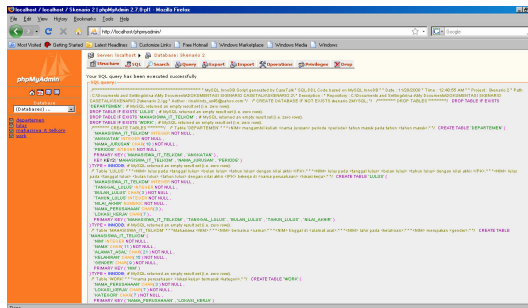
1. Verbalisasi dengan 1 kalimat atau banyak dengan konsep yang sama tidak akan mempengaruhi model yang dihasilkan.
2. Jumlah konsep verbalisasi yang berbeda akan menghasilkan tabel yang berbeda pula.
3. Jumlah konsep verbalisasi yang sama pun tidak menjamin bahwa tabel yang dihasilkan sama. Hal ini disebabkan pemberian *Constraint (Uniqueness Constraint)* pada model.
4. Perubahan urutan verbalisasi dengan konsep yang sama, juga tidak mempengaruhi model yang dihasilkan sebelumnya.

4. HASIL DAN PEMBAHASAN

Setelah DDL *script* dihasilkan dari setiap skenario, maka dapat diimplementasikan ke suatu DBMS. Pada penelitian ini, model akhir dari setiap skenario yang di-generate menjadi *relational database* pada MySQL InnoDB hingga DDL *script* didapatkan, akan diimplementasikan ke dalam DBMS MySQL (dapat dilihat pada Gambar 4-1 dan Gambar 4-2) sehingga akan terbentuk database, tabel, field dan strukturnya.



Gambar 4-1 Implementasi DDL *script* skenario 1

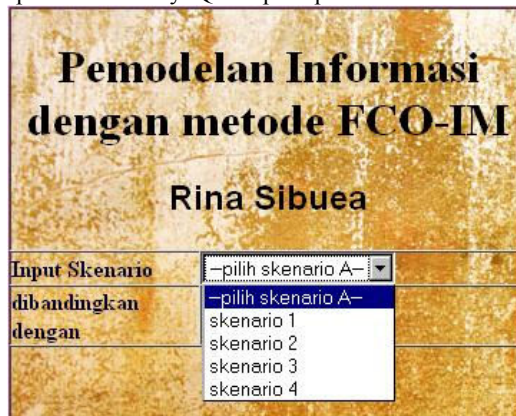


Gambar 4-2 Implementasi DDL *script* skenario 2

Pada penelitian ini, untuk memudahkan dalam hal melihat database, tabel, field dan strukturnya, dibangun sebuah aplikasi berbasis php.

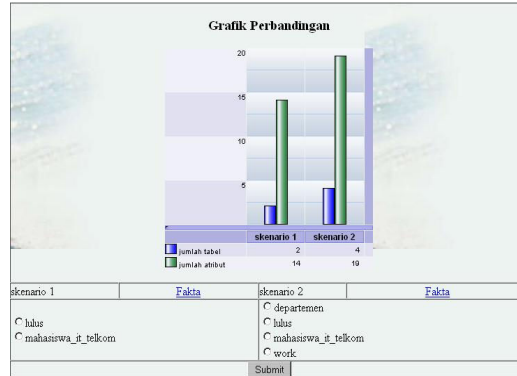
Aplikasi yang dibangun dapat :

1. Melihat database yang ada pada DBMS MySQL seperti pada Gambar 4-3.



Gambar 4-3 Database aktif (tampilan pada Aplikasi)

2. Membandingkan jumlah tabel dan field antar *database*. Dalam hal ini akan dipilih 2 skenario (*database*) yaitu skenario 1 dan 2. Perbandingan dapat dilihat dalam bentuk grafik batang seperti pada Gambar 4-4.



Gambar 4-4 Grafik perbandingan skenario 1 dan 2

Pada hasil di atas, didapatkan bahwa dari informasi yang didapatkan dari *user*, pemodelan informasi untuk skenario 1 menghasilkan 2 tabel dengan jumlah *field* 14. Kedua tabel tersebut adalah tabel lulus dan mahasiswa_it_telkom. Sementara itu, untuk skenario 2 dihasilkan 4 tabel dengan jumlah *field* 19. Keempat tabel tersebut adalah tabel departemen, lulus, mahasiswa_it_telkom, dan work

- Melihat detail struktur tabel dan *field*. Dalam hal ini akan diperlihatkan struktur tabel dari table yang sama yang terdapat pada kedua skenario. Struktur dapat dilihat pada Gambar 4-5.

skenario 1				skenario 2			
Tabel lulus:				Tabel lulus:			
field	type	null	key	field	type	null	key
MAHASISWA_IT_TELKOM	int(11)	NO	PRI	MAHASISWA_IT_TELKOM	int(11)	NO	PRI
TANGGAL_LULUS	int(11)	NO	PRI	TANGGAL_LULUS	int(11)	NO	PRI
BULAN_LULUS	char(3)	NO	PRI	BULAN_LULUS	char(3)	NO	PRI
TAHUN_LULUS	int(11)	NO	PRI	TAHUN_LULUS	int(11)	NO	PRI
NILAI_AKHIR	decimal(10,0)	NO	PRI	NILAI_AKHIR	decimal(10,0)	NO	PRI
NAMA_PERUSAHAAN	char(3)	YES		NAMA_PERUSAHAAN	char(3)	YES	MUL
LOKASI	char(7)	YES		LOKASI KERJA	char(7)	YES	MUL
key_name				key_name			
key_name	column_name			key_name	column_name		
PRIMARY	MAHASISWA_IT_TELKOM			PRIMARY	MAHASISWA_IT_TELKOM		
PRIMARY	TANGGAL_LULUS			PRIMARY	TANGGAL_LULUS		
PRIMARY	BULAN_LULUS			PRIMARY	BULAN_LULUS		
PRIMARY	TAHUN_LULUS			PRIMARY	TAHUN_LULUS		
PRIMARY	NILAI_AKHIR			PRIMARY	NILAI_AKHIR		
FK1_LULUS	MAHASISWA_IT_TELKOM			FK1_LULUS	MAHASISWA_IT_TELKOM		
				FK2_LULUS	NAMA_PERUSAHAAN		
				FK2_LULUS	LOKASI KERJA		

Gambar 4-5 Struktur tabel lulus pada skenario 1 dan 2

- Pada aplikasi ini, juga disediakan sebuah fitur untuk menampilkan data yang ada pada *database*. Sebagai contoh, berikut ini akan ditampilkan data mahasiswa_it_telkom berdasarkan Nim dan bulan kelulusan (Gambar 4-6 dan hasilnya dapat dilihat pada Gambar 4-7).

Isi tabel

Tampilkan 1

MAHASISWA_IT_TELKOM

TANGGAL_LULUS

BULAN_LULUS

TAHUN_LULUS

NILAI_AKHIR

NAMA_PERUSAHAAN

LOKASI

Gambar 4-6 Kriteria pencarian

Jumlah = 68

MAHASISWA_IT_TELKOM	BULAN_LULUS
113910008	Mar
113910020	Mar
112910184	Mar
613910040	Mar
611910007	Mar
611910027	Mar
113920033	Mar
111920152	Mar
112920009	Mar
613920040	Mar
613920043	Mar
613920044	Mar
613920045	Mar
613920046	Mar
613920047	Mar
613920048	Mar
613920049	Mar
611920031	Mar
611920032	Mar
611920033	Mar
611920034	Mar

Gambar 4-7 Hasil penentuan pilihan kriteria

5. KESIMPULAN

Berdasarkan implementasi informasi yang terdapat pada kedua skenario di atas, maka dapat disimpulkan bahwa :

1. Model informasi FCO-IM meliputi semua informasi yang terdapat pada skenario
2. Penambahan informasi ataupun batasan yang dilakukan (dalam hal ini dari skenario 1), tidak terlalu mempengaruhi model informasi sebelumnya
3. Penambahan informasi ataupun batasan terhadap informasi, pasti akan merubah *fact type* pada IGD FCO-IM dan hal ini juga pasti akan mengubah skema *relational database* yang dihasilkan

6. DAFTAR PUSTAKA

- [1] Bakema, Guido., Zwart,Jan.,Pieter., Lek,Harm.,van.,der. ,2002 , *Fully Communication Oriented-Information Modeling*, HAN University, The Netherlands
- [2] Bakema,Guido., Zwart,Jan.,Pieter.,2006, *Innovative information system modeling and development with FCO-IM*,2006, HAN University, The Netherlands
- [3] Zwart,Jan.,Pieter., Bakema,Guido.,*Advances in FCO-IM (1):Disconnected and Overlapping Object Type Expressions*, HAN University, The Netherlands
- [4] Manoku, Elton., Bakema,Guido., 2005, *Integrated Tool Support for Datawarehouse Design*, HAN University, The Netherlands
- [5] Bakema,Guido., 2002, *Crossing the bridge from FCO-IM to ERM*, HAN University, The Netherlands