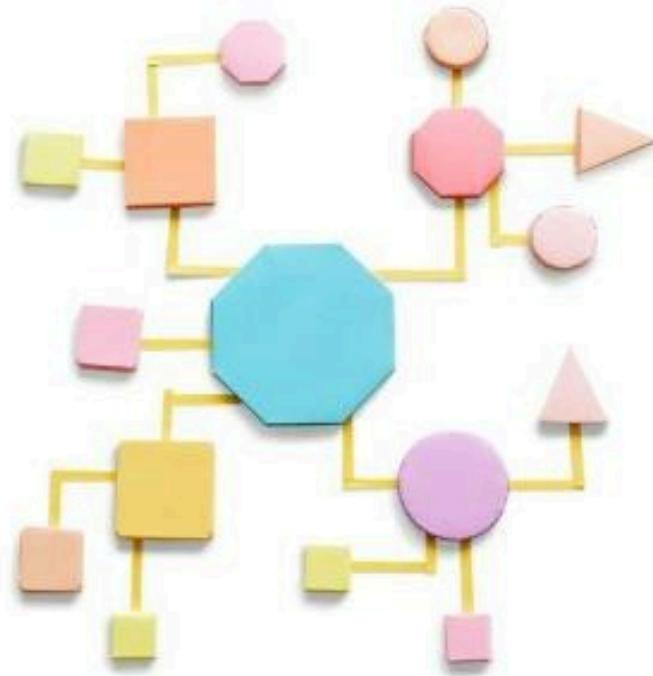


MODUL 6

PEMODELAN DATA SISTEM INFORMASI



PROGRAM STUDI TEKNIK INDUSTRI
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI BANDUNG

2021



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi

DAFTAR ISI

DAFTAR ISI	2
Maklumat Praktikum	3
Tujuan Praktikum.....	4
Tujuan Umum	4
Tujuan Khusus.....	4
Mata Kuliah Acuan	4
<i>Input</i> Praktikum.....	4
<i>Output</i> Praktikum.....	4
Alat & Bahan	4
Posisi Modul 6 PPST III 2021	5
Terminologi Dasar	6
Pendahuluan	7
1. Tahapan Pemodelan Data	8
2. <i>Entity Relationship Diagram</i>	10
3. Normalisasi.....	18
Contoh Kasus Normalisasi	18
Studi Kasus Pemodelan Data	22
Referensi	26
Struktur Laporan	26
Format Laporan.....	27
Format Lembar Pengesahan	28





Maklumat Praktikum



HARAP DIPERHATIKAN

Segala bentuk plagiarisme maupun kecurangan akademis lainnya akan menerima sanksi sesuai peraturan yang berlaku di ITB dan tidak akan kami toleransi.

Aturan dasar:

- Jika sakit/berhalangan untuk mengikuti responsi/praktikum/asistensi yang sifatnya wajib, harap menghubungi dosen koordinator TI3007 PPST III, yakni Bapak Anas Ma'ruf atau Ibu Diss a Riandaso Chandra melalui *e-mail*.
- Jika telah disetujui, harap segera menyerahkan bukti izin yang telah disetujui oleh dosen koordinator (*screenshot e-mail*) ke asistenlsik@gmail.com.



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi



Tujuan Praktikum

Tujuan Umum

- Praktikan memahami prinsip perancangan sistem basis data.
- Praktikan memahami proses perancangan basis data dengan menggunakan *Conceptual Data Model, Logical Data Model, dan Physical Data Model*.
- Praktikan dapat mengimplementasikan model desain basis data yang dirancang dengan menggunakan Microsoft SQL Server.

Tujuan Khusus

Setelah mengikuti praktikum modul ini, praktikan diharapkan mampu:

- Memahami dan mampu merancang *Conceptual Data Model, Logical Data Model, dan Physical Data Model*.
- Mampu merancang desain basis data yang dapat menyelesaikan persoalan manajerial dan operasional.

Mata Kuliah Acuan

- Analisis & Perancangan Sistem Informasi (Pemodelan data)
- Sistem Basis Data

Input Praktikum

- DFD
- *Context Diagram*

Output Praktikum

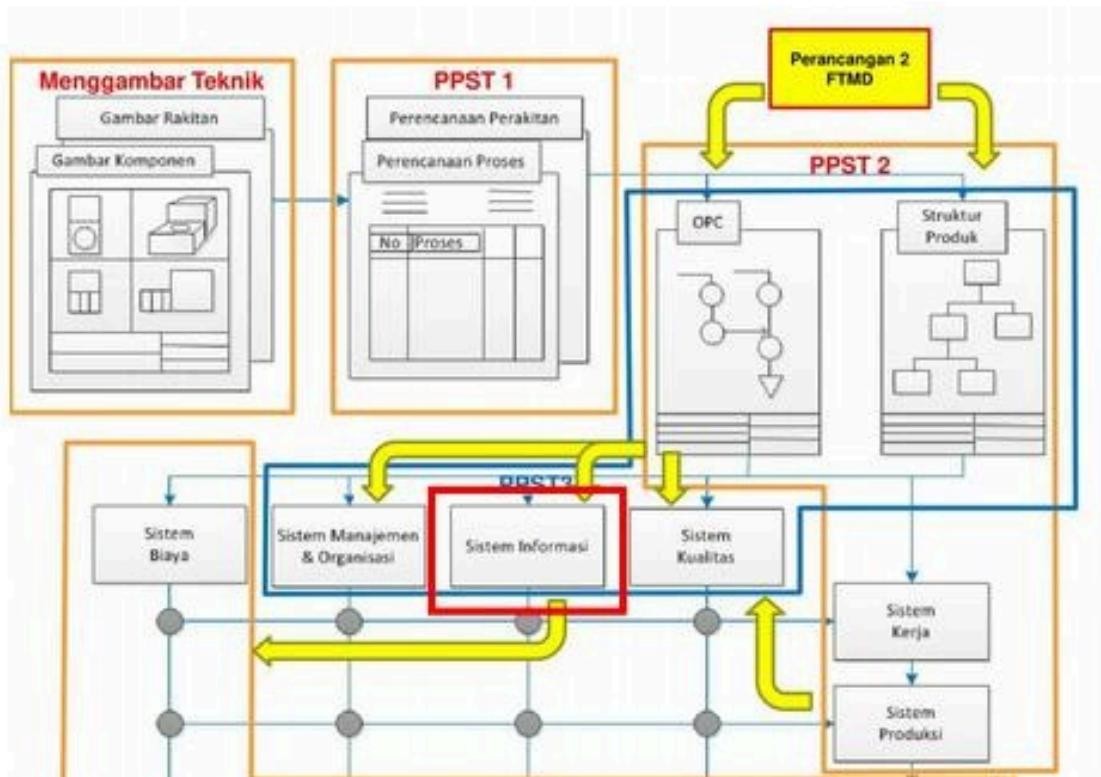
- *Entity Relationship Diagram (ERD)*
- *Logical Data Model, dan Physical Data Model*
- *Database*



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

Posisi Modul 6 PPST III 2021

Posisi Modul 6 Pemodelan Data Sistem Informasi pada rangkaian PPST terdapat pada Gambar 1 yang ditandai dengan kotak berwarna merah.



Gambar 1 Posisi Modul 6 PPST III 2021



5



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

Terminologi Dasar

- **Data**

Data adalah sekumpulan fakta dan statistik yang belum diolah sehingga kurang bermakna. Data-data ini perlu diorganisasikan secara logis agar dapat digunakan oleh stakeholder. Unit data terkecil yang dikenali oleh komputer terdiri dari karakter seperti huruf (A-Z), angka (0-9), dan simbol (?, >, :).

- **Field (Atribut)**

Field adalah unit terkecil dari data yang bermakna dan disimpan dalam sebuah file atau basis data. *Field* dapat berupa kumpulan karakter atau alfabet. Contoh: Nama Mahasiswa, IPK, dan NIM.

- **Record**

Record adalah sekumpulan *field* yang terhubung secara logis dan digunakan untuk mendeskripsikan seseorang, sebuah tempat, atau sesuatu yang lain. Contoh: *record* mahasiswa yang berisikan nama, NIM, alamat, nomor telepon, tanggal kelahiran, dan IPK.

- **File**

File adalah sekumpulan *record* yang saling terkait. Contoh: sebuah *file* terdiri dari *record* semua mahasiswa yang saat ini belajar di Teknik Industri ITB.

- **Database**

Database (basis data) adalah sekumpulan *file* yang memiliki relasi (secara logika) dan didesain untuk memenuhi kebutuhan beberapa pengguna di sebuah organisasi. Contoh: *Database* mahasiswa ITB terdiri dari semua mahasiswa ITB dari berbagai program studi.

- **Database Management System (DBMS)**

DBMS adalah perangkat lunak khusus yang digunakan untuk membuat, mengakses, mengontrol, dan mengelola sebuah *database*. Contoh: MySQL, Microsoft SQL Server, dan Microsoft Access. Dalam praktikum ini, aplikasi DBMS yang akan digunakan adalah Microsoft SQL Server 2012 atau versi lain.

6



TI3007 - Praktikum Perancangan Sistem Terintegrasi III

Modul 6 Pemodelan Data Sistem Informasi

Pendahuluan

Dalam modul sebelumnya (Modul 5 Pemodelan Proses Sistem Informasi) telah dibahas mengenai salah satu metode pengembangan sistem utamanya sistem informasi yaitu dengan siklus SDLC (System Development Life Cycle). Pada SDLC, terdapat tahap analisis kebutuhan dan design yang terdiri dari perancangan pemodelan proses dan data. Jika pada modul sebelumnya telah dibahas mengenai pemodelan proses maka pada modul ini akan dibahas lebih lanjut terkait pemodelan data.

Pemodelan data merupakan teknik yang digunakan untuk mengorganisasikan dan mendokumentasikan data-data dari suatu sistem. Model data dibuat dalam bentuk grafik yang menggambarkan struktur data dalam sistem nyata beserta karakteristik, relasi, batasan-batasan, serta transformasi data. Model ini akan digunakan antara lain sebagai fasilitas komunikasi/interaksi antara sistem *designer*, *application programmer*, dan *end users*.



1. Tahapan Pemodelan Data

Dalam melakukan pemodelan data terdapat beberapa tahapan yang harus dilakukan agar model data dapat dibuat dengan baik dan benar. Adapun tahapan tersebut dapat dilihat sebagai berikut.

Tabel 1 Tahapan Pemodelan Data

Tahapan	Langkah	Implementasi
Perancangan Model Konseptual	1. Analisis data dan kebutuhan Langkah ini dibutuhkan untuk menentukan: <ul style="list-style-type: none"> • Informasi yang dibutuhkan, informasi yang diharapkan akan dihasilkan sistem, dan informasi apa yang tersedia di sistem <i>existing</i>. • Pengguna informasi, yang mencakup siapa yang menggunakan informasi dan bagaimana informasi tersebut akan digunakan oleh pengguna informasi. • Sumber informasi, yang mencakup dimana dan bagaimana informasi tersebut diambil. 	Pada dasarnya, terdapat beberapa cara untuk memodelkan data secara konseptual seperti dalam bentuk <i>Hierarchical Database Model</i> dan <i>Relational Database Model</i> . Pada modul ini, akan digunakan <i>Relational Database Model</i> dengan

	<p>2. Perbaikan dan hubungan dan normalisasi informasi, atribut yang dimiliki entitas dan relasi yang dimiliki antar data.</p> <p>Verifikasi model data dibutuhkan untuk memastikan sistem dapat memenuhi seluruh kebutuhan sistem. Model data harus bersifat kohesif, yang berarti masing-masing entitas memiliki hubungan yang kuat, dan model harus lengkap dan komprehensif.</p> <p>4. Perancangan basis data Software DBMS dipilih dengan mempertimbangkan biaya, fitur software DBMS, portability, dan hardware yang dibutuhkan DBMS terkait.</p>	<p>• Relationship diagram</p> <ul style="list-style-type: none"> • Requeriments dalam model • Relationship diagram • Relationship diagram
Pemilihan DBMS	Memilih model <i>database management system</i> (DBMS)	Penentuan penggunaan <i>software SQL Server</i> sebagai perangkat DBMS (<i>database management system</i>)



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

Tahapan	Langkah	Implementasi
Logical Design	<p>1. Memetakan model konseptual menjadi model <i>logical</i>. Pada tahap ini, entitas diperlakukan menjadi tabel-tabel.</p> <p>2. Melakukan validasi model <i>logical</i> dengan normalisasi. Normalisasi adalah proses transformasi menjadi struktur data yang lebih kecil dan stabil. Penjelasan lebih lanjut terkait normalisasi akan dijelaskan pada bagian selanjutnya.</p> <p>3. Melakukan validasi model <i>logical</i> dengan integrity constraints. Pada sistem, mungkin terdapat data memiliki atribut yang harus diisi dengan input tertentu. Sebagai contoh, terdapat entitas Kelas yang memiliki atribut Jam Kelas. Jam Kelas harus diisi dengan nilai di antara 7 pagi hingga 5 sore. Validasi model memastikan model memenuhi <i>integrity constraint</i> tersebut.</p> <p>4. Melakukan validasi model <i>logical</i> dengan membandingkan dengan <i>user requirements</i>.</p>	<p>Pembuatan <i>Logical Database Model (LDM)</i></p> <ul style="list-style-type: none"> • Model data yang mentranslasikan atau mendokumentasikan entitas-entitas yang ada pada model konseptual ke dalam model data yang didukung DBMS. • Dibuat oleh <i>system analyst</i> dan atau <i>system designer</i>
Physical Design	<p>1. Menentukan data base organisasi. Langkah ini mencakup menentukan data apa saja yang perlu disimpan, sehingga dapat diketahui ukuran <i>storage</i> yang diperlukan untuk <i>database</i>. Frekuensi penggunaan masing-masing <i>data store</i> juga perlu diidentifikasi.</p> <p>2. Menentukan peraturan keamanan dan integritas Pada langkah ini, dilakukan penentuan <i>database role</i>, yaitu peranan-peranan dan keuntungan yang dimiliki oleh suatu kelompok user tertentu. Dilakukan juga penetapan pengaturan keamanan.</p> <p>3. Menentukan ukuran performansi</p>	<p>Pembuatan <i>Physical Data Model (PDM)</i></p> <ul style="list-style-type: none"> • Mendeskripsikan struktur fisik data. • PDM menampilkan atribut serta tipe data atribut yang terdapat dalam entitas serta ukuran <i>storage</i> yang dibutuhkan. • PDM dibuat oleh <i>system designer</i>.

2. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan salah satu model yang paling umum digunakan untuk melakukan perancangan model data konseptual. Pada modul ini, pembuatan ERD dilakukan menggunakan **Crow's Foot Database Notation** pada Microsoft Visio 2016/365.

Pada pembuatan ERD menggunakan Microsoft Visio 2016/365, terdapat sejumlah simbol yang digunakan seperti di bawah ini:

- **Entitas**

Sebuah entitas dapat berupa orang, objek, event atau konsep di dalam lingkungan pengguna di dalam organisasi yang datanya akan disimpan. Penggambaran entitas pada Microsoft Visio dapat dilihat pada Gambar 2.

Contoh:

- Orang : Pegawai, mahasiswa, pasien
- Tempat : Negara, provinsi
- Objek : Mesin, ruangan, bangunan, pabrik
- Event : Registrasi, penjualan, pembelian
- Konsep : Course, account



Gambar 2 Entitas pada Microsoft Visio

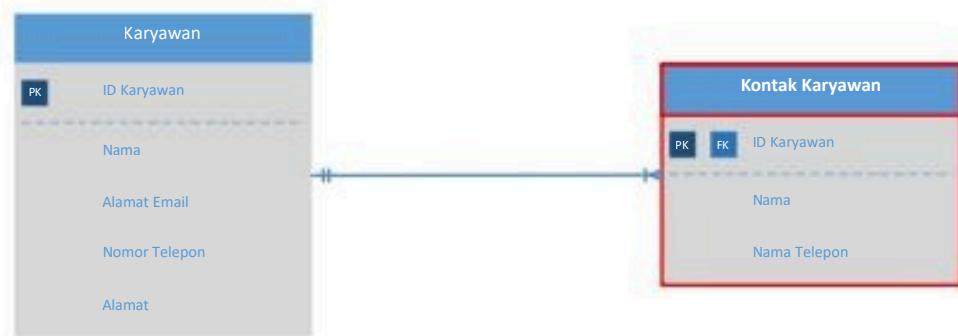
- **Child Entity**

Child entity merupakan entitas yang memiliki sifat *existence-dependent* dimana keberadaan (*existence*) entitas ini dependen/bergantung terhadap keberadaan entitas lain yang biasa disebut *parent entity*. *Child entity* dapat memiliki *primary key* yang sama dengan *primary key* dari *parent entity* namun jika dibutuhkan dapat memiliki *primary key*-nya sendiri. Tentunya karena bersifat *existence-dependent* maka *child entity* harus terhubung dengan *parent entity*-nya. *Child entity* memiliki simbol berupa kotak bergaris ganda.

Contohnya: Terdapat entitas kontak karyawan berisi ID Karyawan, Nama, dan Nomor Telepon yang merupakan *child entity* dari entitas karyawan. Penggambaran kedua entitas ini dapat dilihat pada Gambar 3.



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi



Gambar 3 Parent-child entity

- **Atribut**

Atribut adalah properti atau karakteristik dari sebuah entitas yang berguna (untuk diketahui atau disimpan) bagi organisasi. Penempatan atribut dapat dilihat pada Gambar 4.



Gambar 4 Atribut

Contoh:

Entitas: Mahasiswa

Atribut:

- NIM
- Nama
- Alamat
- Nomor Telepon
- Tanggal Mulai Kuliah
- IPK

- Status Pembayaran

Gambaran dari entitas Mahasiswa beserta atribut-atributnya dapat dilihat pada Gambar 5.



11



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi



Gambar 5 Entitas dan Atribut

- Relasi

Relasi menggambarkan asosiasi antarentitas. Jenis-jenis relasi sebagai berikut:

- a. Berdasarkan sifatnya, relasi dapat dibedakan menjadi 3 yaitu

 1. *One to one*
 2. *One to many*
 3. *Many to many*

Terdapat beberapa notasi yang dapat digunakan ketika menggambarkan relasi antarentitas, pada praktikum ini notasi yang akan digunakan yaitu Crow's Foot Notation. Contoh dari relasi yang digambarkan menggunakan notasi Crow's Foot dapat dilihat pada [Error! Reference source not found.](#)

Pada relasi yang menggunakan Crow's Foot, jumlah asosiasi terdapat dua indikator. Indikator pertama menunjukkan jumlah asosiasi maksimal entitas. Sementara itu, indikator kedua menunjukkan jumlah asosiasi minimal entitas. Berikut dijelaskan lebih lanjut mengenai simbol dan penjelasan indikator pada relasi dengan Crow's Foot.

12

Tabel 2 *Crow's Foot Notation*



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

(many)

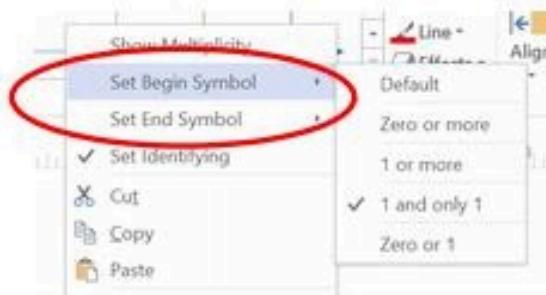


Jumlah asosiasi minimal adalah nol



Jumlah asosiasi minimal adalah satu

Pada Microsoft Visio 2016/365, relasi antar entitas dapat diubah dengan melakukan klik kanan pada relasi, lalu pilih *Set Begin Symbol* dan *Set End Symbol*, lalu pilih simbol Crow's Foot yang sesuai. Simbol yang dapat dipilih berupa *zero or more*, *1 or more*, *1 and only 1*, dan *zero or 1*.



Gambar 6 Penetapan simbol relasi pada Microsoft Visio

Untuk memahami lebih lanjut mengenai jenis-jenis relasi berdasarkan sifat serta perbedaannya, berikut contoh dari masing-masing jenis relasi.

- One to one.

Pada suatu kantor terdapat entitas karyawan dan entitas komputer. Hubungan antara kedua entitas ini yaitu bahwa karyawan dapat dipinjamkan satu buah komputer atau tidak sama sekali dan satu buah komputer dapat dipinjamkan untuk satu karyawan atau



13



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

tidak sama sekali. Oleh karena itu, hubungan antarentitas ini yaitu *one on one* dengan jumlah asosiasi maksimum keduanya yaitu satu dan jumlah asosiasi minimumnya yaitu nol. Maka simbol yang digunakan yaitu simbol zero or 1 pada kedua sisi.

- One to many.

Terdapat entitas Marketing Team dan Marketing Project. Marketing Team harus melakukan satu atau lebih Marketing Project sedangkan Marketing Project hanya dapat dilakukan oleh satu Marketing Team dan pasti harus dilakukan oleh satu Marketing Team. Oleh karena itu, hubungan antarentitas ini adalah *one to many* dimana untuk entitas Marketing ke Marketing Project jumlah asosiasi maksimumnya adalah *many* dan minimumnya adalah satu sedangkan untuk entitas Marketing Project ke Marketing jumlah asosiasi maksimumnya adalah *one* dan minimumnya adalah *one*. Maka, simbol yang digunakan yaitu 1 and only 1 (dari Marketing Project ke Marketing Team) dan 1 or more (dari Marketing Team ke Marketing Project). Penggambaran dari entitas Marketing Team dan Marketing Project beserta relasinya dapat dilihat pada Gambar 7.



Jumlah Anggota

FK Objektif
Tanggal Mulai

Tanggal Pembentukan

Deskripsi
Tanggal SelesaiGambar 7 Contoh relasi *one to many* antarentitas

- Many to many.

Terdapat entitas Karyawan dan Project. Karyawan dapat memilih untuk melakukan projek (baik 1 maupun lebih) atau tidak sama sekali sedangkan Project harus dikerjakan oleh minimal satu karyawan. Oleh karena itu, kedua entitas ini memiliki hubungan *many to many*, dimana untuk entitas Karyawan ke Project jumlah asosiasi minimum nya yaitu nol dan jumlah asosiasi maksimumnya yaitu banyak (*many*) sedangkan untuk Project ke Karyawan jumlah asosiasi minimumnya yaitu satu dan jumlah asosiasi maksimumnya yaitu banyak (*many*). Maka simbol yang digunakan adalah zero or more (Karyawan ke Project) dan 1 or more (Project ke Karyawan).

- b. Berdasarkan tingkatannya, relasi dapat dibagi menjadi 3, yaitu:



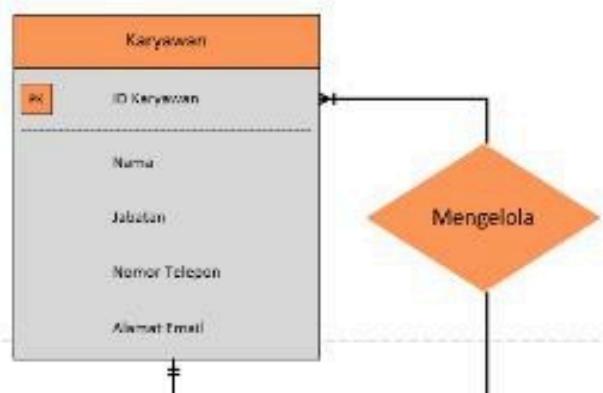
14



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

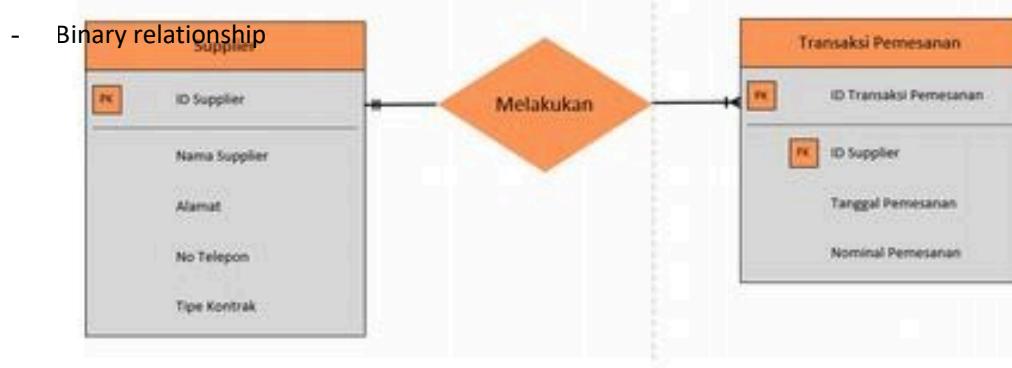
- Unary relationship

Unary relationship adalah relasi antara satu entitas.



Gambar 8 Unary relationship

Binary relationship adalah relasi antara dua entitas.



Gambar 9 *Binary relationship*

- Ternary relationship

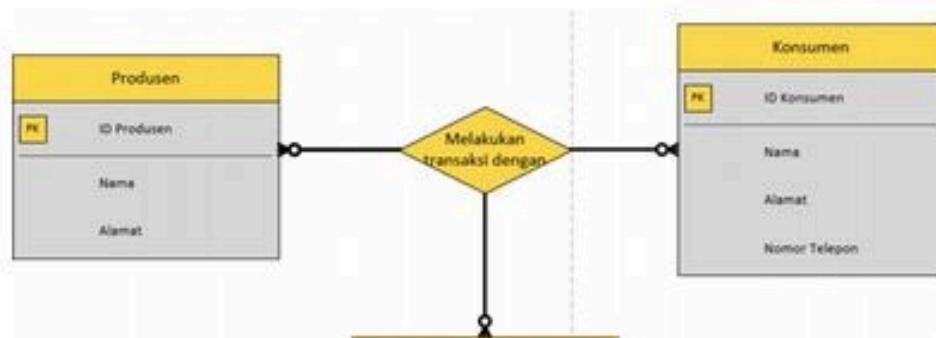
Ternary relationship adalah relasi antara lebih dari dua entitas.



15



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi





Gambar 10 *Ternary relationship*

Ketika membentuk suatu relasi dalam pemodelan data, maka terdapat atribut-atribut yang menjadi acuan dalam menghubungkan entitas-entitas yang ada. Atribut-atribut ini disebut sebagai *primary key* dan *foreign key*. Berikut merupakan penjelasan mengenai *primary key* dan *foreign key*.

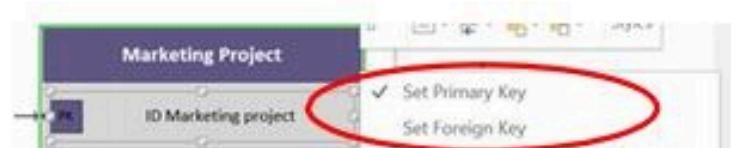
- ***Primary Key***

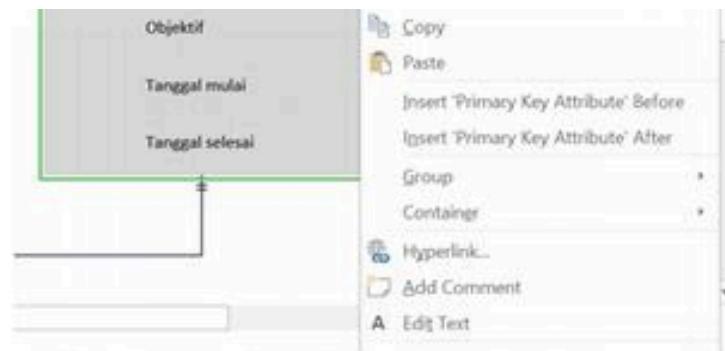
Primary key merupakan suatu nilai dalam basis data yang digunakan untuk mengidentifikasi suatu baris dalam tabel. Nilai ini bersifat unik untuk setiap *record* dan umumnya digunakan sebagai identitas suatu entitas. Hal ini berarti Anda tidak dapat menambahkan 2 record dengan *primary key* yang sama.

- ***Foreign Key***

Foreign key merupakan suatu atribut yang berperan sebagai penghubung antar entitas yang berkaitan. *Foreign key* melengkapi hubungan suatu entitas terhadap *primary key* yang berkaitan dengan entitas induknya. Jika sebuah *primary key* terhubung dengan entitas lainnya, maka keberadaan *primary key* pada entitas tersebut disebut sebagai *foreign key*.

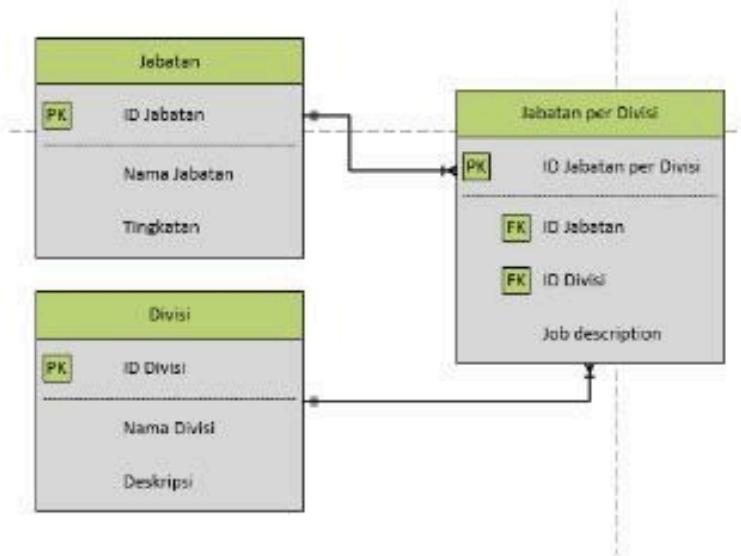
Pada Microsoft Visio, penetapan suatu atribut sebagai *primary key* dan *foreign key* dilakukan dengan melakukan klik kanan pada atribut tabel entitas, lalu memilih "Set Primary Key" atau "Set Foreign Key".





Gambar 11 Pengaturan *primary key* dan *foreign key* pada Microsoft Visio

Berikut ini merupakan contoh penggunaan *primary key* dan *foreign key* pada ERD.



Gambar 12 Contoh penggunaan *primary key* dan *foreign key*

3. Normalisasi

Normalisasi merupakan teknik yang digunakan untuk memastikan sebuah *database* terbebas dari redundansi data. Redundansi data yaitu suatu kondisi ketika terdapat sejumlah informasi sama yang tersimpan di dalam *database*. Redundansi data dapat menimbulkan banyak permasalahan. Berikut merupakan permasalahan yang dapat disebabkan oleh redundansi data.

- *Data inconsistency (lack of data integrity)*

- *Data anomalies*

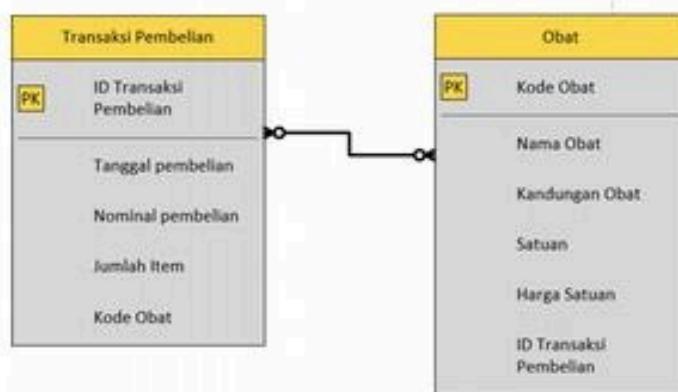
Data anomalies dapat dibedakan menjadi tiga, yaitu:

- *Modification anomalies*: suatu kondisi dimana data tidak konsisten terjadi karena adanya modifikasi
- *Insertion anomalies*: suatu kondisi dimana data tidak konsisten terjadi karena adanya penambahan data
- *Deletion anomalies*: suatu kondisi dimana data tidak konsisten terjadi karena adanya penghapusan data

Umumnya, normalisasi dilakukan untuk entitas-entitas yang memiliki relasi *many to many* karena relasi ini banyak menimbulkan redundansi pada entitas-entitas yang bersangkutan.

Contoh Kasus Normalisasi

Diberikan dua buah entitas yaitu Transaksi Pembelian dan Obat. Kedua entitas ini memiliki hubungan *many to many* karena satu obat dapat dibeli dalam banyak pembelian dan pada satu transaksi pembelian dapat terbeli banyak obat. Hubungan ini dapat dilihat pada gambar di bawah ini:



Gambar 13 Relasi antara entitas Obat dan Transaksi Pembelian

Jika dilihat dari Tabel 3 dan Tabel 4, terdapat dua *record* yang memiliki informasi yang sama atau disebut juga dengan redundansi data.



Tabel 3 Transaksi Pembelian

ID Transaksi Pembelian	Tanggal Pembelian	Nominal Pembelian	Kode Obat	Jumlah Item
301	4 Januari 2018	Rp2.000	1	1
303	12 Januari 2018	Rp10.000	2	1
312	14 Januari 2018	Rp10.000	3	1
315	21 Januari 2018	Rp28.000	3	2
315	21 January 2018	Rp28.000	4	1

Tabel 4 Obat

Kode Obat	Nama Obat	Kandungan Obat	Satuan	Harga Satuan	ID Transaksi Pembelian
1	Panadol	Parasetamol	Strip	Rp2.000	301
2	Antimo	Dimenhidrinat	Sachet	Rp5.000	303
3	OBH Combi	Succus liquiritiae, Ammonium chloride	Botol	Rp10.000	312
3	OBH Combi	Succus liquiritiae, Ammonium chloride	Botol	Rp10.000	315
4	Mastin	Antosianin	Box	Rp8.000	315

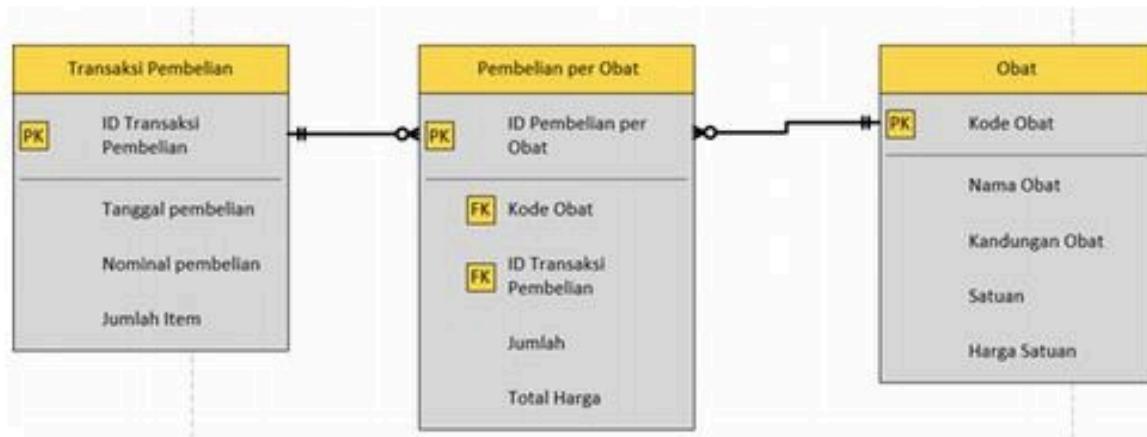
Redundansi data seperti yang terlihat pada Tabel 3 dan Tabel 4 tidak efisien dengan alasan sebagai berikut:

- Menyebabkan pemborosan pada memori.
- Menyebabkan kesulitan untuk melakukan *update* data pada setiap tabelnya karena harus dikerjakan dua kali pada tabel yang berbeda.
- Peluang kesalahan pada data semakin besar jika hal ini terjadi (contoh: bisa jadi saat *update* lupa untuk melakukan *update* di salah satu tabel sehingga terjadi *discrepancy*).

Untuk menghindari hal tersebut, maka dilakukanlah metode normalisasi dengan cara membuat entitas baru di antara dua entitas yang memiliki hubungan *many to many* seperti gambar di bawah ini:



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi



Gambar 14 Hasil Normalisasi

Entitas hasil normalisasi akan memiliki hubungan *many to one* dengan entitas-entitas asalnya. Pada kasus Transaksi dan Obat, hasil tabel normalisasi yaitu Pembelian per Obat akan memiliki hubungan *many to one* dengan Transaksi Pembelian dan Obat. Hal ini dikarenakan satu pembelian per obat hanya terdapat di satu transaksi dan hanya meliputi satu jenis obat saja. Berikut merupakan tabel hasil normalisasi yang telah dibuat.

Tabel 5 Transaksi Pembelian Hasil Normalisasi

ID Transaksi Pembelian	Tanggal Pembelian	Nominal Pembelian	Jumlah Item
301	4 Januari 2018	Rp2.000	1
303	12 Januari 2018	Rp10.000	1
312	12 Januari 2018	Rp10.000	1
315	21 Januari 2018	Rp28.000	3

Tabel 6 Obat Hasil Normalisasi

Kode Obat	Nama Obat	Kandungan Obat	Satuan	Harga Satuan
1	Panadol	Parasetamol	Strip	Rp2.000
2	Antimo	Dimenhidrinat	Sachet	Rp10.000
3	OBH Combi	Succus liquiritiae, Ammonium chloride	Botol	Rp10.000
4	Mastin	Antosianin	Box	Rp10.000



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi

Tabel 7 Pembelian per Obat Hasil Normalisasi

ID Pembelian per Obat	ID Transaksi Pembelian	Kode Obat	Jumlah	Total Harga
A1	301	1	1	Rp2.000
A2	303	2	2	Rp10.000
A3	312	3	1	Rp10.000
A4	315	3	2	Rp20.000
A5	315	4	1	Rp8.000



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

Studi Kasus Pemodelan Data

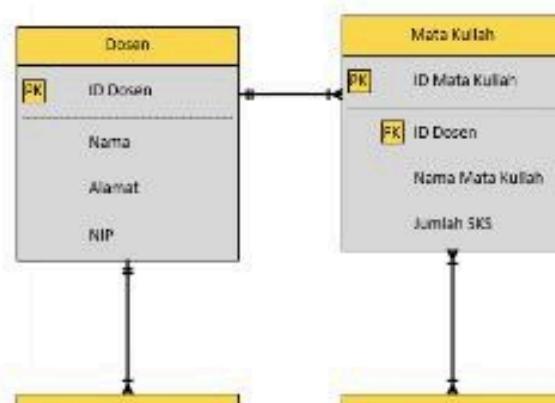
Sebuah sekolah tinggi yang baru saja berdiri ingin mengimplementasikan sebuah sistem informasi untuk memudahkan para mahasiswanya dalam melakukan pengisian rencana studi. Dengan sistem informasi yang berbasis web ini, mahasiswa sekolah tinggi tersebut dapat memilih berbagai macam mata kuliah yang tersedia sesuai dengan ketentuan yang berlaku. Setelah mahasiswa mengambil suatu mata kuliah, sistem akan menghubungkan mata kuliah yang diambil tersebut dengan dosen pengajar dan ruangan kelasnya. Namun karena sekolah tinggi yang baru saja didirikan ini masih mengalami sejumlah masalah internal, terdapat sejumlah ketentuan yang perlu dipertimbangkan dalam pembuatan sistem. Adapun ketentuan tersebut adalah sebagai berikut

- Setiap mahasiswa akan memiliki dosen wali. Tugas dosen wali ini adalah melakukan verifikasi atas rencana studi yang dipilih oleh mahasiswa dengan ketentuan sekolah tinggi.
- Karena keterbatasan jumlah dosen, seorang dosen dapat mengajar sejumlah mata kuliah namun setiap mata kuliah hanya akan diampu oleh satu dosen.

Berikut adalah contoh hasil pembuatan ERD, LDM, dan PDM berdasarkan studi kasus diatas.

ERD

Berikut merupakan ilustrasi ERD sederhana untuk pengisian rencana studi *online*



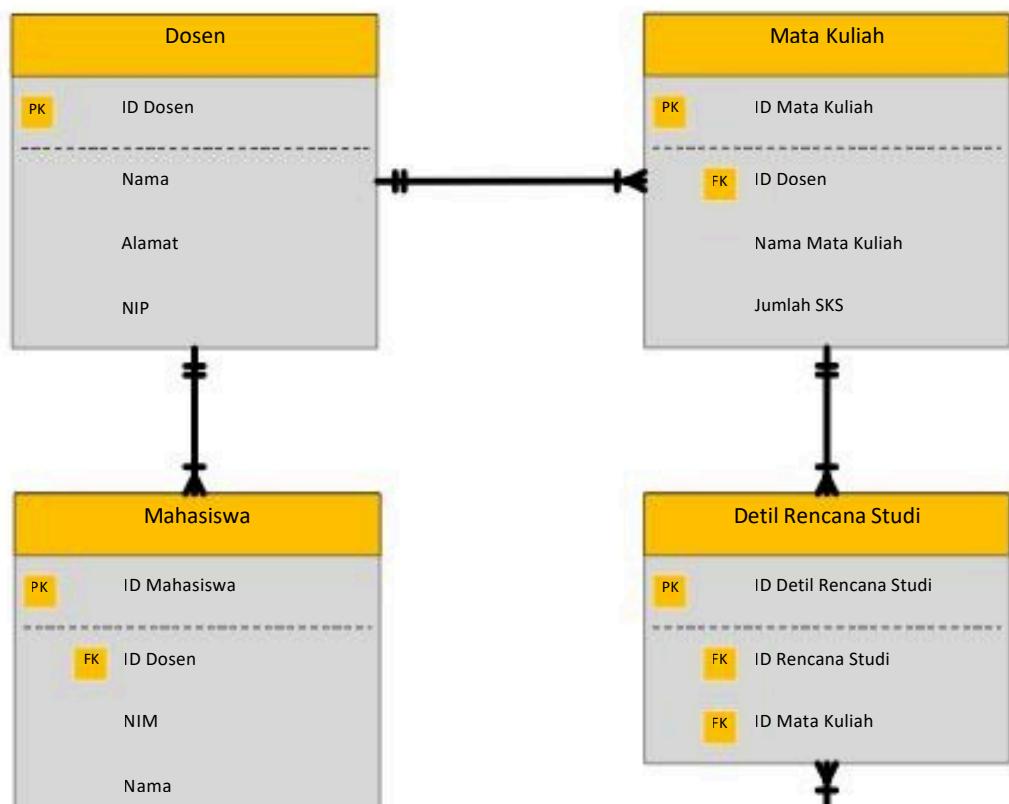
Gambar 15 ERD Pengisian Rencana Studi

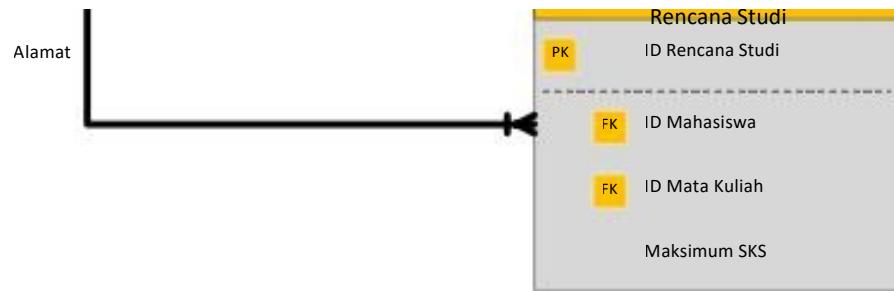
22



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi

Pada gambar di atas dapat dilihat adanya hubungan many to many antara entitas Rencana Studi dan Mata Kuliah sehingga perlu dilakukan normalisasi. Berikut adalah hasil normalisasi yang dilakukan.





Gambar 16 Normalisasi ERD Pengisian Rencana Studi

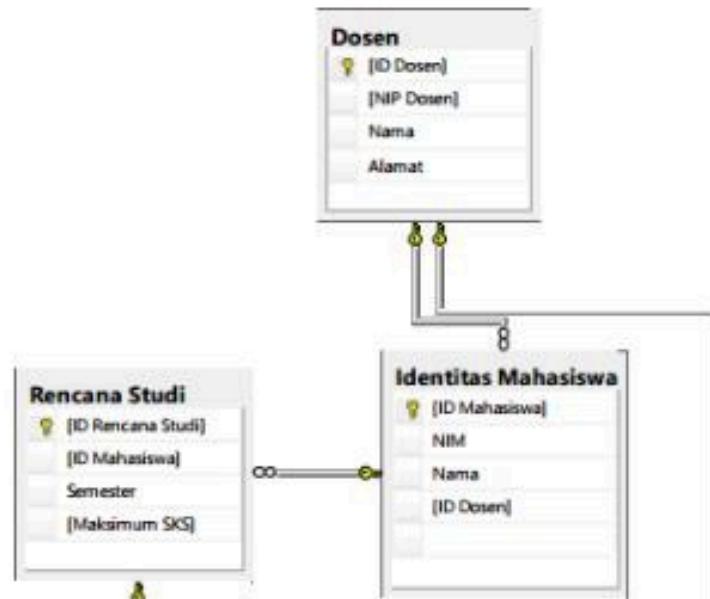
23

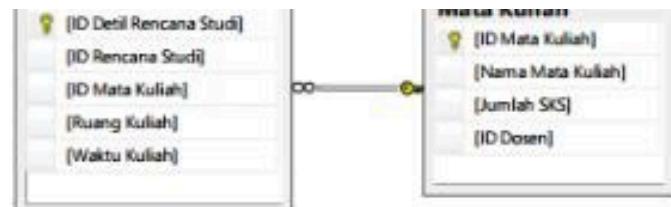


TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi

LDM

Berikut merupakan LDM yang diperoleh dari Microsoft SQL Server 2012





Gambar 17 LDM Pengisian Rencana Studi

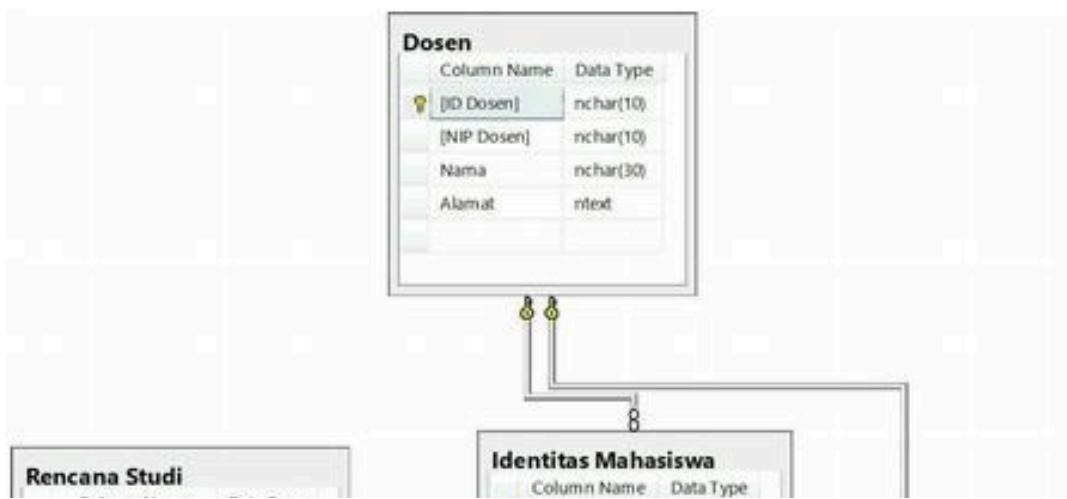
PDM

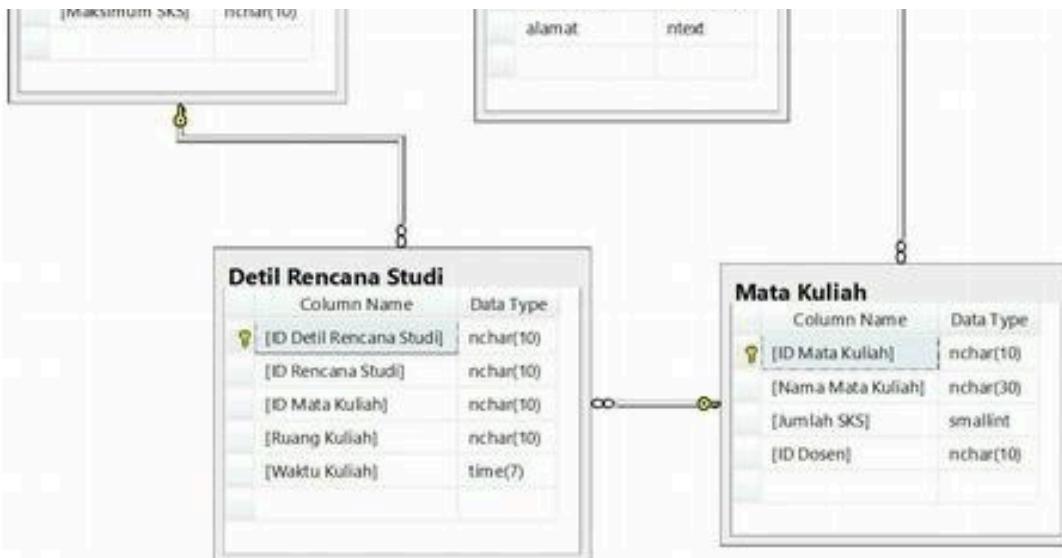
Berikut merupakan PDM yang diperoleh dari Microsoft SQL Server 2012

24



TI3007 - Praktikum Perancangan Sistem Terintegrasi III Modul 6 Pemodelan Data Sistem Informasi





Gambar 18 PDM Pengisian Rencana Studi

25



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi

Referensi

- Coronel, C., Morris, S., & Rob, P. (2011). *Database Systems: Design, Implementation, and Management 9th Edition*. Boston: Cengage Learning.
- Whitten, J. L., & Bentley, L. D. (2007). *Systems Analysis and Design Methods 7th Edition*. New York: McGraw-Hill.

Struktur Laporan

Lembar Pengesahan

Lembar Asistensi

Daftar Isi

Daftar Gambar

Daftar Tabel

Bab 1 Pendahuluan

 1.1 Latar Belakang

 1.2 Tujuan

 1.3 *Flowchart*

Bab 2 Pengolahan Data

Bab 3 Analisis

Bab 4 Kesimpulan & Saran

 4.1 Kesimpulan

 4.2 Saran

 4.2.1. Saran untuk Praktikum

 4.2.2. Saran untuk Asisten

Daftar Pustaka

Lampiran

26



TI3007 - Praktikum Perancangan Sistem Terintegrasi III
Modul 6 Pemodelan Data Sistem Informasi

Format Laporan

Ukuran A4. Margin KIRI-ATAS-KANAN-BAWAH: 3-2-2-2

Font:

- Isi Laporan Calibri 10
- Judul dan sub judul Cambria 12
- Spasi multiple 1.15

Cover laporan sama dengan PPST 2

Header kiri : Modul 6 PPST 3: Pemodelan Data Sistem Informasi

Header kanan : Nama asisten/NIM

Footer kiri : NIM anggota kelompok (NIM-nya saja)

Footer kanan : Nomor halaman



27





Format Lembar Pengesahan

LEMBAR PENGESAHAN

Asisten Laboratorium Sistem Informasi dan Keputusan ITB (LSIK ITB) yang bertandatangan di bawah ini menyatakan bahwa Laporan Praktikum Perancangan Sistem Terintegrasi (PPST) III Modul ..., yang disusun oleh:

1. Nama Anggota 1 (NIM Anggota 1)
2. Nama Anggota 2 (NIM Anggota 2)
3. Nama Anggota 3 (NIM Anggota 3)

telah dapat dikumpulkan pada:

Hari:

Tanggal:

Waktu:

Bandung, dd/mm/yyyy

Nama Asisten
(NIM Asisten)

