

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1. Alat dan Bahan

Dalam perancangan ini dibutuhkan beberapa alat dan bahan sebagai penunjang keberhasilan perancangan *database* Sistem *Monitoring* Perizinan Perkuliahan Mahasiswa.

3.1.1. Hardware

Dalam perancangan *database* ini, perangkat keras (*hardware*) yang digunakan adalah laptop dengan spesifikasi:

1. *Processor intel core-i3.*
2. RAM 8 GB.
3. *Harddisk 500 GB.*
4. Monitor, *keyboard*, dan *mouse*.

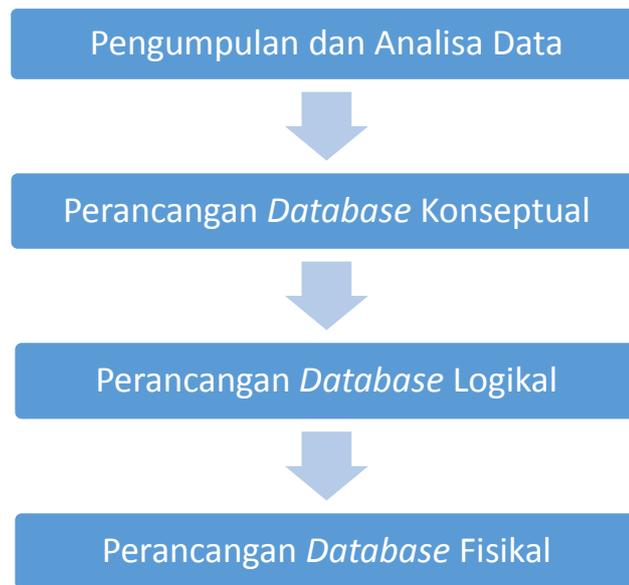
3.1.2 Software

Selain perangkat keras (*hardware*), dibutuhkan juga perangkat lunak (*software*) yang digunakan untuk mendukung perancangan *database* Sistem *Monitoring* Perizinan Perkuliahan Mahasiswa, yaitu:

1. Sistem operasi *windows 8.1.*
2. *Tools* sebagai pendukung perancangan *database* yaitu *XAMPP v3.2.2*, *Microsoft Visio 2013*, *PHPMyAdmin*
3. *MySQL* sebagai *Database Management System (DBMS)*

3.2. Metode Perancangan *Database*

Dalam perancangan *database* ini meliputi 4 metode tahapan yang dapat dilihat pada gambar 3.1.



Gambar 3.1 Metode Perancangan *Database*

Penjelasan metode perancangan *database* pada gambar 3.1 sebagai berikut:

Pada perancangan ini meliputi 4 tahapan yaitu *Requirement Collection and Analysis* (Pengumpulan dan Analisa Data), *Conceptual Database Design* (Perancangan *Database* Konseptual), *Logical Database Design* (Perancangan *Database* Logikal), dan *Physical Database Design* (Perancangan *Database* Fisikal).

3.3. Metode Pengumpulan Data

Dalam perancangan ini ada dua sumber yang dapat digunakan untuk mengumpulkan data, yaitu sumber data primer dan sumber data sekunder. Data primer adalah data yang didapat dengan cara mengamati secara langsung objek

yang diteliti. Sumber data primer yang dilakukan dalam penelitian ini adalah wawancara atau tanya jawab dengan mendapatkan data-data secara langsung dari sumber yang mengerti sehubungan dengan pengamatan, penulis bertanya langsung dengan pihak terkait yaitu mahasiswa jurusan TI Universitas Muhammadiyah Yogyakarta. Data dari hasil wawancara dapat dilihat di lampiran 1. Sedangkan data sekunder adalah data yang diperoleh melalui data yang telah diteliti dan dikumpulkan oleh pihak lain yang berkaitan dengan permasalahan penelitian seperti dokumentasi dari staff layanan jurusan TI UMY.

3.4. Perancangan *Database* Konseptual

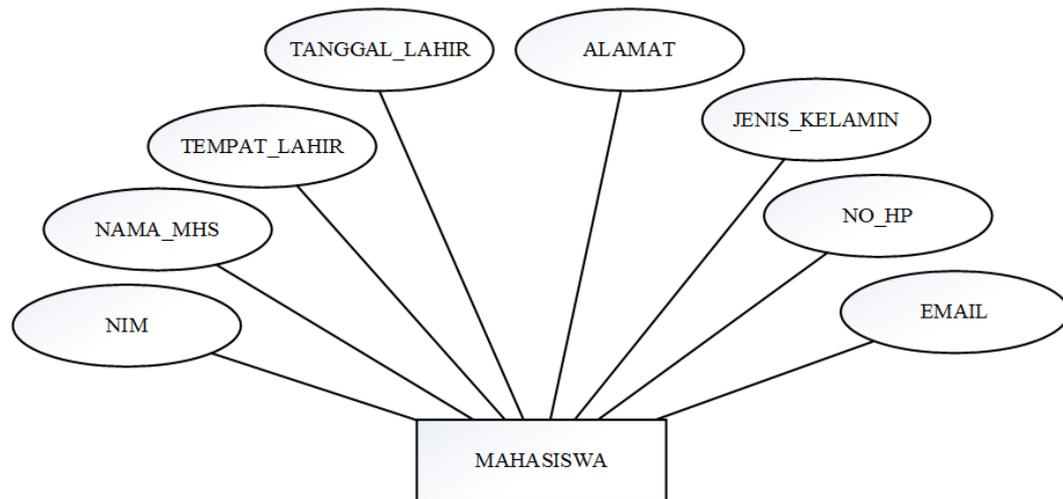
Perancangana *Database* Konseptual merupakan proses membangun suatu model berdasarkan informasi yang digunakan oleh perusahaan atau organisasi, tanpa pertimbangan perencanaan fisik. Dalam merancang *database* konseptual terdapat beberapa langkah, antara lain mengidentifikasi entitas (*entity*), mengidentifikasi atribut dari suatu entitas, menentukan *primary key*, mengidentifikasi *relationship* atau hubungan satu entitas dengan entitas yang lain.

3.4.1. Entitas

Pada tahap ini menentukan dan mengidentifikasi entitas yang akan digunakan. Entitas didapatkan berdasarkan hasil wawancara dan data yang telah dikumpulkan dari dokumentasi yang ada. Berikut adalah entitas dan *attributes* dari sistem *monitoring* perizinan perkuliahan mahasiswa.

1. Entitas Mahasiswa

Entitas mahasiswa terdiri dari 8 *attributes* di antaranya nim, nama_mhs, tempat lahir, tanggal lahir, alamat, jenis kelamin, no_hp, dan *email*. *Attributes* tersebut dapat dilihat pada gambar 3.2.



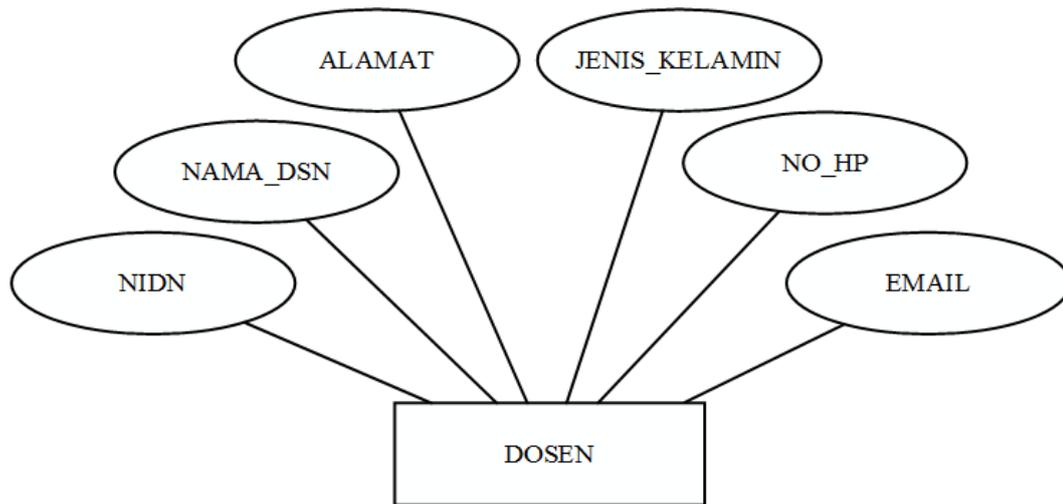
Gambar 3.2 Entitas Mahasiswa

Penjelasan entitas mahasiswa pada gambar 3.2 adalah sebagai berikut :

Pada *attributes* nim, nama_mhs, tempat_lahir, tanggal_lahir, alamat, jenis_kelamin, no_hp, *email* perlu dimasukkan ke dalam entitas mahasiswa, karena berfungsi untuk mengetahui informasi data mahasiswa.

2. Entitas Dosen

Entitas dosen terdiri dari 6 *attributes* di antaranya nidn, nama_dsn, alamat, jenis kelamin, no_hp, dan *email*. *Attributes* tersebut dapat dilihat pada gambar 3.3.



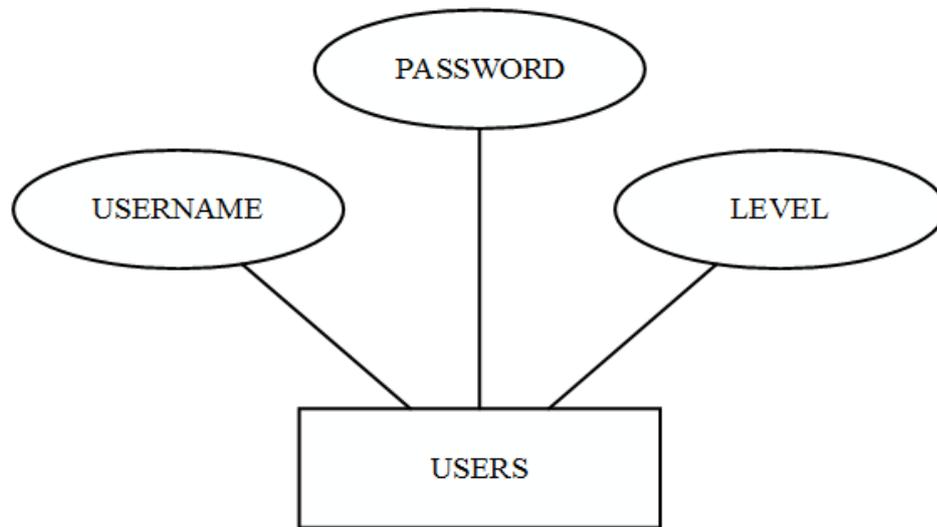
Gambar 3.3 Entitas Dosen

Penjelasan entitas dosen pada gambar 3.3 adalah sebagai berikut :

Pada *attributes* *nidn*, *nama_dsn*, *alamat*, *jenis_kelamin*, *no_hp*, *email* perlu dimasukkan ke dalam entitas dosen, karena berfungsi untuk mengetahui informasi data dosen.

3. Entitas *Users*

Entitas *users* terdiri dari 3 *attributes* di antaranya *username*, *password*, dan *level*. *Attributes* tersebut dapat dilihat pada gambar 3.4.



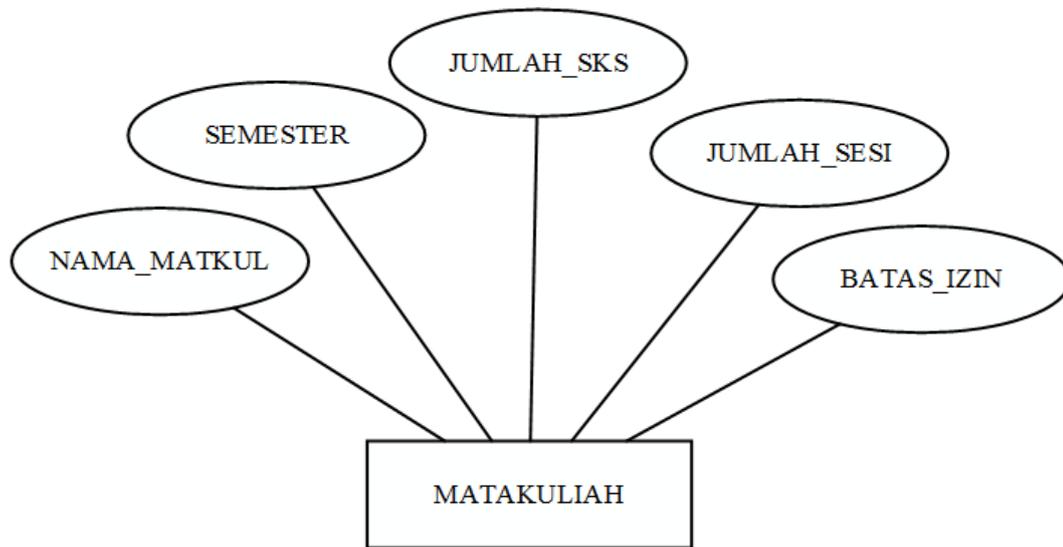
Gambar 3.4 Entitas *Users*

Penjelasan entitas *users* pada gambar 3.4 adalah sebagai berikut :

Atribut *level* perlu dimasukkan ke dalam entitas *users* berfungsi untuk mengetahui *level* dari kedua *users* yang ada dalam *database* yaitu mahasiswa dan dosen. Dengan adanya atribut *level*, *database* dapat membagi *level* sesuai dengan fungsi dari masing-masing *users*. Sedangkan *username* dan *password* perlu dimasukkan ke dalam entitas *users*, karena berfungsi untuk *login users*.

4. Entitas Matakuliah

Entitas matakuliah terdiri dari 5 *attributes* di antaranya nama_matakuliah, semester, jumlah_sks, jumlah_sesi, dan batas_izin. *Attributes* tersebut dapat dilihat pada gambar 3.5.



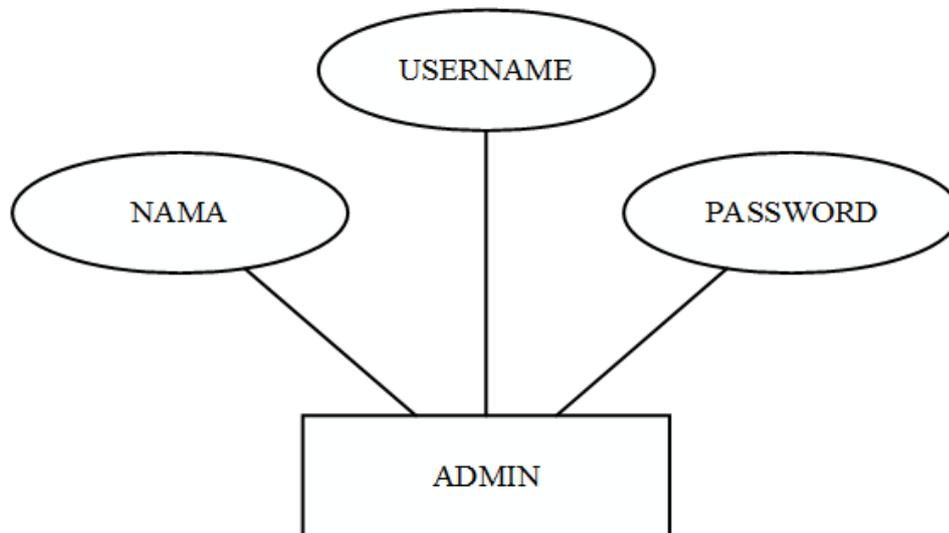
Gambar 3.5 Entitas Matakuliah

Penjelasan entitas matakuliah pada gambar 3.5 adalah sebagai berikut :

Pada *attributes* nama_matkul, semester perlu dimasukkan ke dalam entitas matakuliah, karena berfungsi untuk mengetahui informasi data matakuliah. Pada *attributes* jumlah_sks, perlu dimasukkan ke dalam entitas matakuliah karena berfungsi untuk mengetahui total sks pada suatu matakuliah. Pada *attributes* jumlah_sesi, perlu dimasukkan ke dalam entitas matakuliah karena berfungsi untuk mengetahui total pertemuan pada suatu matakuliah. Sedangkan *attributes* batas_izin perlu dimasukkan ke dalam entitas matakuliah, karena berfungsi untuk mengetahui berapa kali mahasiswa dapat melakukan perizinan pada suatu matakuliah.

5. Entitas Admin

Entitas admin terdiri dari 3 *attributes* di antaranya nama, *username*, dan *password*. *Attributes* tersebut dapat dilihat pada gambar 3.6.



Gambar 3.6 Entitas Admin

Penjelasan entitas admin pada gambar 3.6 adalah sebagai berikut :

Pada *attributes* nama perlu dimasukkan ke dalam entitas admin, karena berfungsi untuk mengetahui nama admin. Sedangkan *username* dan *password* perlu dimasukkan ke dalam entitas admin, karena berfungsi untuk *login* admin.

3.4.2. *Primary Key*

Pada tahap ini menentukan dan mengidentifikasi *primary key* dari setiap entitas. *Primary key* didapatkan dengan cara mengidentifikasi setiap atribut yang ada. Atribut dapat dikatakan sebagai *primary key* apabila atribut tersebut unik atau tidak ada data yang duplikat. Berikut adalah *primary key* dari sistem *monitoring* perizinan perkuliahan mahasiswa.

1. *Primary Key* Entitas Mahasiswa

Pada entitas mahasiswa memiliki beberapa *attributes* sebagai berikut:

- a. NIM
- b. Nama_Mhs

- c. Tempat_Lahir
- d. Tanggal_lahir
- e. Alamat
- f. Jenis_kelamin
- g. No_hp
- h. *Email*

Berdasarkan pada daftar *attributes* di atas, pada *attributes* nim bersifat unik karena setiap mahasiswa mempunyai nim yang berbeda. Oleh karena itu *attributes* nim akan menjadi *primary key* pada entitas mahasiswa.

2. *Primary Key* Entitas Dosen

Pada entitas dosen memiliki beberapa *attributes* sebagai berikut:

- a. NIDN
- b. Nama_Dsn
- c. Alamat
- d. Jenis_kelamin
- e. No_hp
- f. *Email*

Berdasarkan pada daftar *attributes* di atas, *attributes* nidn bersifat unik karena setiap dosen mempunyai nidn yang berbeda. Oleh karena itu *attributes* nidn akan menjadi *primary key* pada entitas dosen.

3. *Primary Key* Entitas *Users*

Pada entitas *users* memiliki beberapa *attributes* sebagai berikut:

- a. *Username*

b. *Password*

c. *Level*

Berdasarkan pada daftar *attributes* di atas, tidak ada *attributes* yang bersifat unik, maka perlu ditambahkan satu *attributes* lagi dengan nama *Id_user*, *attributes* *Id_user* ini yang akan menjadi *primary key* dari entitas *users*.

4. *Primary Key* Entitas Matakuliah

Pada entitas matakuliah memiliki beberapa *attributes* sebagai berikut:

a. *Nama_Matkul*

b. *Semester*

c. *Jumlah_SKS*

d. *Jumlah_Sesi*

e. *Batas_Izin*

Berdasarkan pada daftar *attributes* di atas, tidak ada *attributes* yang bersifat unik, maka perlu ditambahkan satu *attributes* lagi dengan nama *Kode_Matkul*, *attributes* *Kode_Matkul* ini yang akan menjadi *primary key* dari entitas matakuliah.

5. *Primary Key* Entitas Admin

Pada entitas admin memiliki beberapa *attributes* sebagai berikut:

a. *Nama*

b. *Username*

c. *Password*

Berdasarkan pada daftar *attributes* di atas, tidak ada *attributes* yang bersifat unik, maka perlu ditambahkan satu *attributes* lagi dengan nama *Id_admin*, *attributes* *Id_admin* ini yang akan menjadi *primary key* dari entitas *admin*.

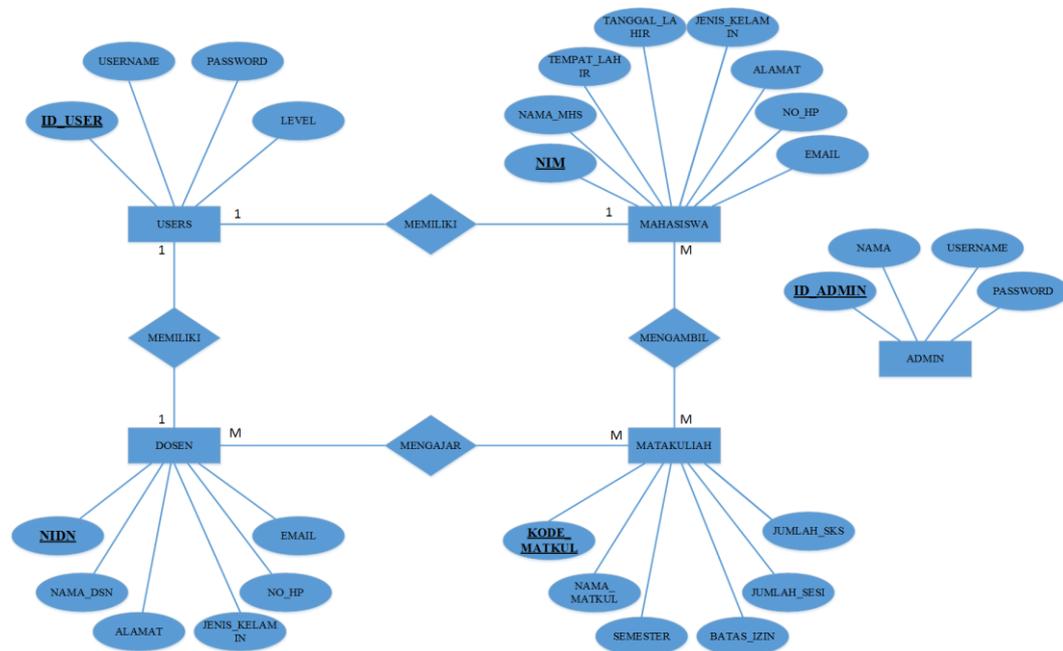
3.4.3. Sistem yang akan dijalankan

Sistem yang akan dijalankan diartikan sebagai sistem yang akan digunakan pada sistem *monitoring* perizinan perkuliahan mahasiswa di Universitas Muhammadiyah Yogyakarta. Berikut adalah perancangan sistem yang akan dijalankan.

1. Setiap mahasiswa hanya dapat memiliki satu akun *user* yang digunakan untuk *login*.
2. Setiap dosen hanya dapat memiliki satu akun *user* yang digunakan untuk *login*.
3. Mahasiswa dapat mengambil lebih dari satu matakuliah dan satu matakuliah dapat diambil oleh banyak mahasiswa.
4. Mahasiswa dapat melakukan perizinan perkuliahan yang setiap matakuliah memiliki batas izin akan tetapi mahasiswa masih dapat melakukan perizinan walaupun melebihi batas izin dari suatu matakuliah.
5. Dosen dapat mengajar lebih dari satu matakuliah dan satu matakuliah dapat diajar lebih dari satu dosen.
6. Dosen hanya dapat melihat data perizinan mahasiswa sesuai matakuliah yang diajarkan.
7. Dosen hanya dapat memberikan info tugas sesuai dengan matakuliah yang diajarkan.

3.4.4. ERD (Entity Relationship Diagram)

Dalam sistem *monitoring* perizinan perkuliahan mahasiswa yang dibuat terdapat beberapa entitas yaitu entitas mahasiswa, entitas dosen, entitas matakuliah, entitas *users*, dan entitas admin. Dari semua entitas tersebut membentuk sebuah ERD diagram. ERD diagram dapat dilihat pada gambar 3.7.



Gambar 3.7 Entity Relationship Diagram

Penjelasan ERD pada gambar 3.7 adalah sebagai berikut :

1. Entitas mahasiswa mempunyai relasi *one-to-one* dengan entitas *users*, artinya satu mahasiswa hanya dapat memiliki satu *user*.
2. Entitas dosen mempunyai relasi *one-to-one* dengan entitas *users*, artinya satu dosen hanya dapat memiliki satu *user*.
3. Entitas mahasiswa memiliki relasi *many-to-many* dengan entitas matakuliah yang artinya mahasiswa dapat mengambil lebih dari satu matakuliah dan

matakuliah dapat diambil lebih dari satu mahasiswa, maka relasi ini akan menciptakan tabel baru.

4. Entitas dosen memiliki relasi *many-to-many* dengan entitas matakuliah yang artinya dosen dapat mengajar lebih dari satu matakuliah dan matakuliah dapat diajar lebih dari satu dosen, maka relasi ini akan menciptakan tabel baru.
5. Entitas admin tidak memiliki relasi dengan entitas manapun.

3.5. Perancangan Database Logikal

Perancangan *database level* logikal merupakan tahapan untuk memetakan proses perancangan konseptual kedalam model *database* yang akan digunakan yaitu pemetaan dari ERD (*entity relationship diagram*) ke bentuk tabel. Model yang digunakan dalam perancangan *database* ini adalah model relasi.

3.5.1. Konversi ERD ke dalam Tabel

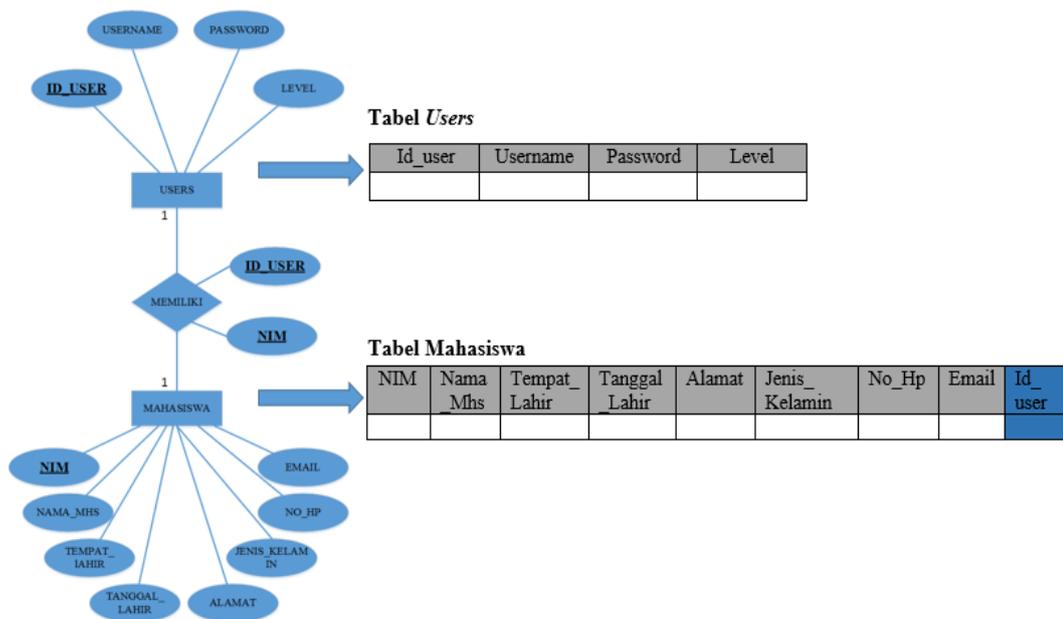
Konversi ERD ke dalam bentuk tabel adalah proses memasukan entitas menjadi nama tabel dan *attribute* menjadi nama kolom beserta relasinya ke dalam sebuah tabel yang saling berhubungan.

1. Konversi dengan relasi 1:1 (*one to one*)

Konversi entitas yang memiliki relasi 1:1 (*one to one*) yaitu menghubungkan dua entitas yang akan direpresentasikan dalam bentuk penambahan atau penyertaan atribut-atribut ke tabel yang mewakili salah satu dari kedua entitas tersebut. Pada ERD yang ditunjukkan pada gambar 3.7 ditunjukkan terdapat relasi *one to one* yaitu antara entitas *users* dengan entitas mahasiswa dan entitas *users* dengan entitas dosen. Dari entitas-entitas tersebut akan

direpresentasikan kedalam tabel, dapat dilihat pada gambar 3.8 dan gambar 3.9 sebagai berikut.

a. Entitas *Users* dengan Entitas Mahasiswa

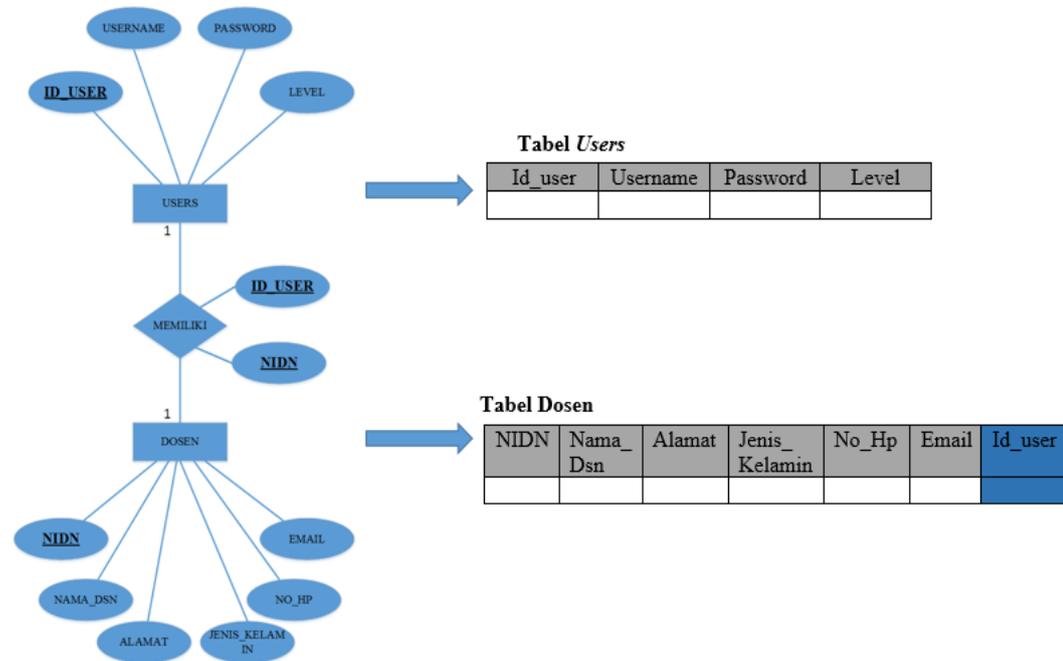


Gambar 3.8 Konversi Relasi antara Entitas *Users* dengan Entitas Mahasiswa

Penjelasan konversi relasi entitas *users* dengan entitas mahasiswa pada gambar 3.8 adalah sebagai berikut:

Entitas mahasiswa memiliki relasi *one to one* dengan entitas *users* yang artinya satu mahasiswa memiliki satu *user* maka kolom *Id_user* yang ada pada tabel *users* perlu ditambahkan ke tabel mahasiswa yang berfungsi sebagai *foreign key* pada tabel mahasiswa.

b. Entitas *Users* dengan Entitas Dosen



Gambar 3.9 Konversi Relasi antara Entitas *Users* dengan Entitas Dosen

Penjelasan konversi relasi antara entitas *users* dengan entitas dosen pada gambar 3.9 adalah sebagai berikut:

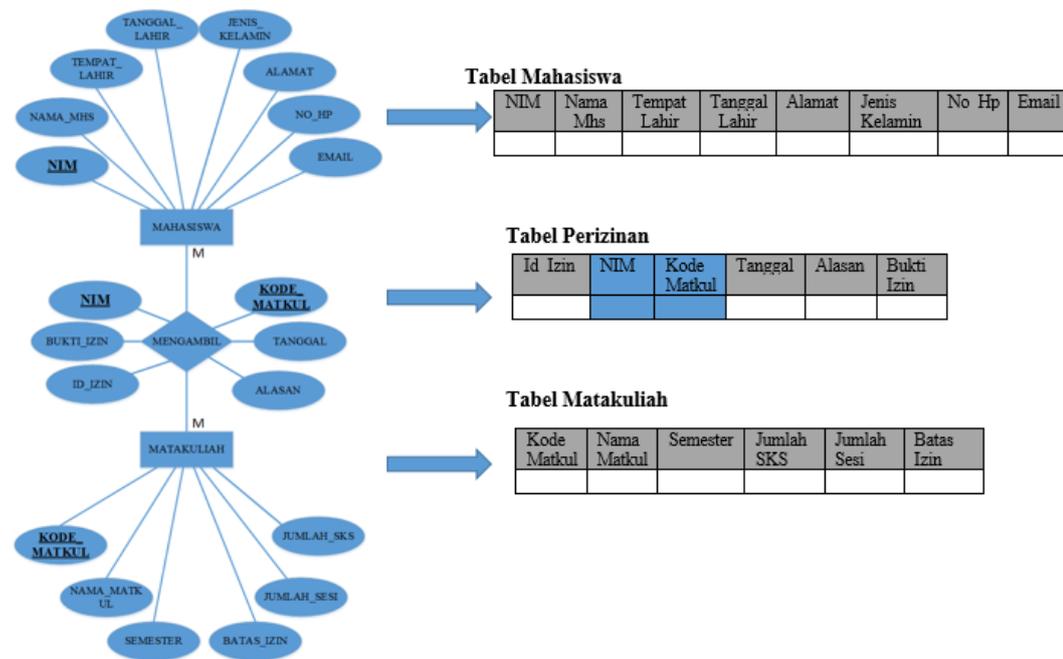
Entitas dosen memiliki relasi *one to one* dengan entitas *users* yang artinya satu dosen memiliki satu *user* maka kolom *Id_user* yang ada pada tabel *users* perlu ditambahkan ke tabel dosen yang berfungsi sebagai *foreign key* pada tabel dosen.

2. Konversi dengan relasi M:M (*many to many*)

Konversi entitas yang memiliki relasi M:M (*many to many*) yaitu menghubungkan dua entitas, diwujudkan dalam bentuk tabel (*file data*) khusus yang memiliki *field (foreign key)* yang berasal dari kunci-kunci (*key*) dari entitas yang dihubungkan sehingga menciptakan tabel baru. Pada ERD yang ditunjukkan pada gambar 3.7 ditunjukkan terdapat relasi *many to many* yaitu antara entitas

mahasiswa dengan entitas matakuliah dan entitas dosen dengan entitas matakuliah. Dari entitas-entitas tersebut akan direpresentasikan kedalam tabel, dapat dilihat pada gambar 3.10 dan gambar 3.11 sebagai berikut:

a. Entitas Mahasiswa dengan Entitas Matakuliah



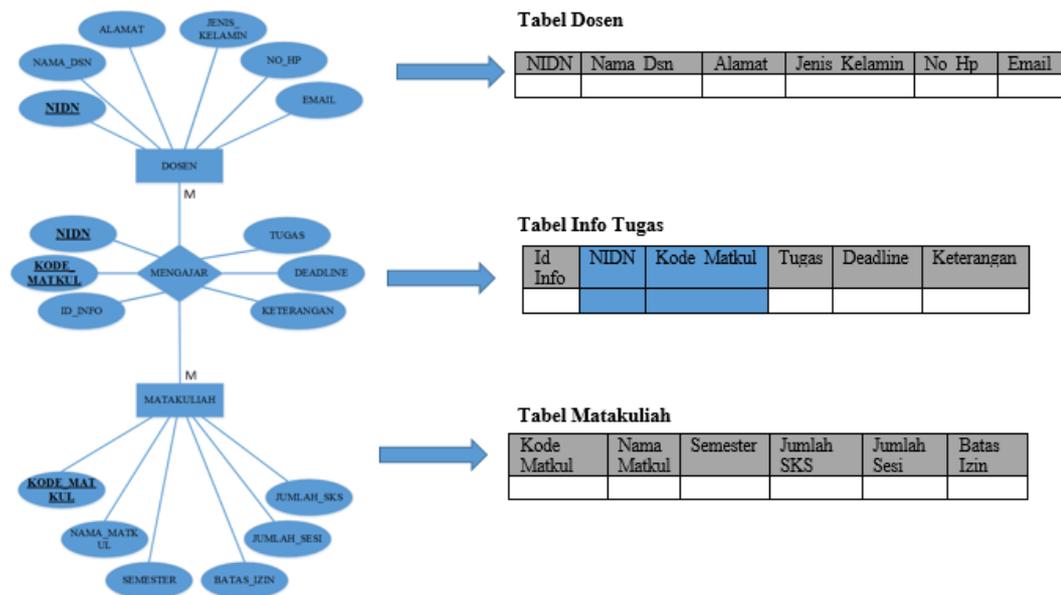
Gambar 3.10 Konversi Relasi antara Entitas Mahasiswa dengan Entitas Matakuliah

Penjelasan konversi relasi antara entitas mahasiswa dengan entitas matakuliah pada gambar 3.10 adalah sebagai berikut:

Entitas mahasiswa memiliki relasi *many to many* dengan entitas matakuliah sehingga tercipta tabel baru yang menghubungkan entitas mahasiswa dengan matakuliah yang diberi nama tabel perizinan, dimana kolom NIM pada tabel mahasiswa dan kolom Kode_Matkul pada tabel matakuliah ditambahkan ke dalam tabel perizinan sebagai *foreign key*, maka perlu ditambahkan Id_izin sebagai

primary key dan kolom lainnya sebagai penambahan informasi seperti tanggal, alasan, dan bukti_izin.

b. Entitas Dosen dengan Entitas Matakuliah



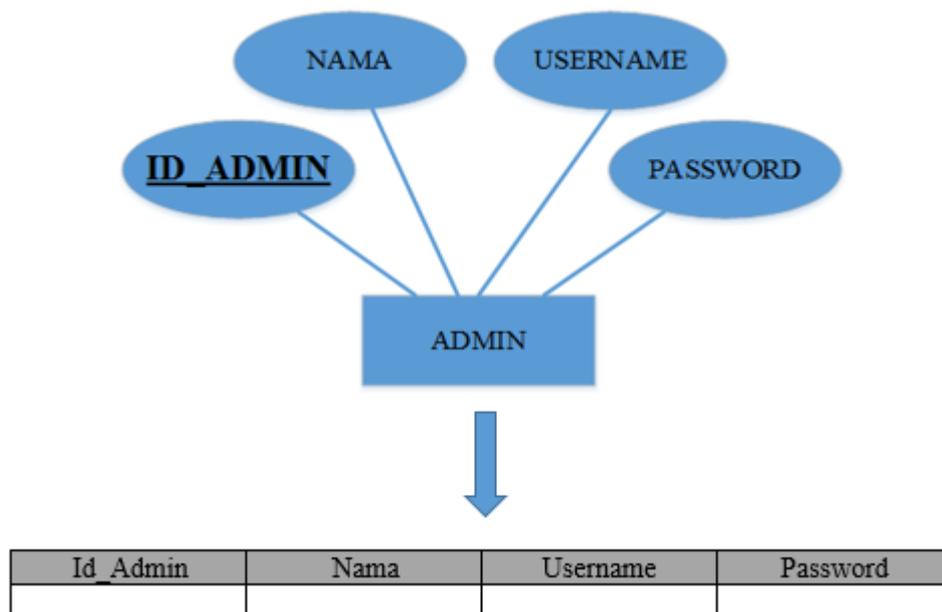
Gambar 3.11 Konversi Relasi antara Entitas Dosen dengan Entitas Matakuliah

Penjelasan konversi relasi antara entitas dosen dengan entitas matakuliah pada gambar 3.11 adalah sebagai berikut:

Entitas dosen memiliki relasi *many to many* dengan entitas matakuliah sehingga tercipta tabel baru yang menghubungkan entitas dosen dengan matakuliah yang diberi nama tabel info tugas, dimana kolom NIDN pada tabel dosen dan Kode_Matkul pada tabel matakuliah ditambahkan ke dalam tabel info tugas sebagai *foreign key*, maka perlu ditambahkan Id_info sebagai *primary key* dan kolom lainnya sebagai penambahan informasi seperti tugas, *deadline*, dan keterangan.

3. Konversi entitas kuat

Konversi entitas kuat adalah entitas yang dilibatkan atau dicantumkan secara jelas dalam ERD, dan tidak memiliki ketergantungan terhadap entitas lain. Pada ERD yang ditunjukkan pada gambar 3.7 ditunjukkan terdapat entitas kuat yaitu entitas admin yang dapat dilihat pada gambar 3.12.



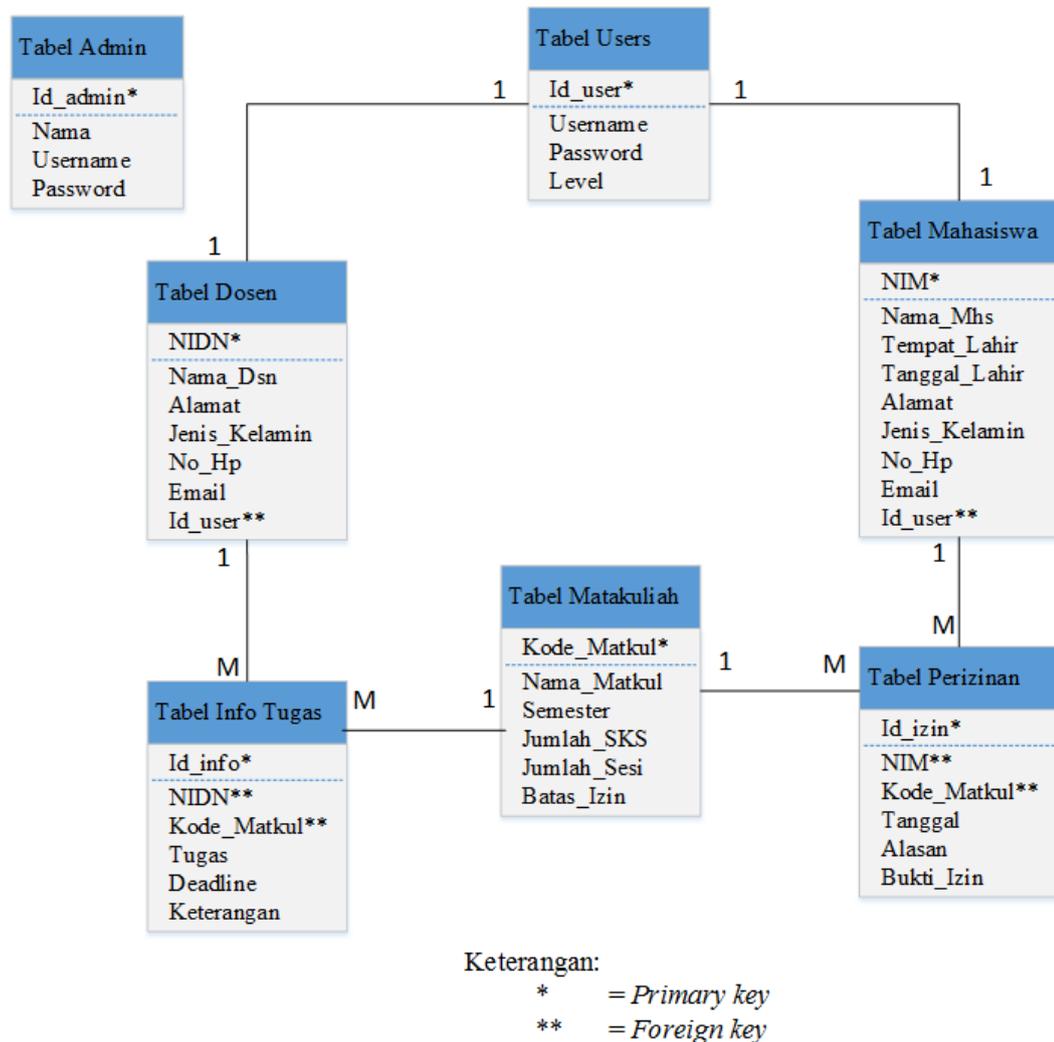
Gambar 3.12 Konversi Entitas Kuat

Penjelasan konversi entitas kuat pada gambar 3.12 adalah sebagai berikut:

Pada proses konversi entitas admin ke dalam bentuk tabel pada gambar 3.12 dijelaskan bahwa entitas menjadi nama tabel dan *attribute* menjadi nama kolom.

3.5.2. Relasi antar Tabel

Setelah dilakukan konversi ERD ke dalam tabel maka akan didapatkan relasi antar tabel yang dapat dilihat pada gambar 3.13 sebagai berikut.



Gambar 3.13 Relasi antar Tabel

3.5.3. Kamus Data Database

Kamus data (*data dictionary*) merupakan salah satu komponen kunci dari DBMS yang mencakup informasi mengenai struktur *database*. Setiap elemen data yang disimpan dalam *database* memiliki catatan kamus data yang mendeskripsikan elemen-elemen tersebut. Kamus data sangat membantu analis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu

dapat dilakukan dengan lengkap dan terstruktur. Berikut adalah kamus data dari sistem *monitoring* perizinan perkuliahan mahasiswa.

1. Kamus Data Mahasiswa

Tabel 3.1 Kamus Data Tabel Mahasiswa

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
NIM	Char	11	<i>Primary Key</i>	-
Nama_Mhs	Varchar	20	-	-
Tempat_Lahir	Varchar	15	-	-
Tgl_Lahir	Date	-	-	-
Alamat	Text	-	-	-
Jenis_Kelamin	Enum	-	-	-
No_Hp	Varchar	12	-	-
Email	Varchar	35	-	-
Id_user	Int	-	<i>Foreign Key</i>	<i>Users</i>

Penjelasan kamus data mahasiswa pada tabel 3.1 adalah sebagai berikut:

- a. NIM adalah kolom yang menunjukkan nomor induk mahasiswa dari masing-masing mahasiswa, dengan tipe data *Char* karena pada kolom NIM berisi data dengan panjang yang tetap. Kolom NIM memiliki ukuran panjang data 11 karena merupakan panjang data dari nomor induk mahasiswa yang ada di Universitas Muhammadiyah Yogyakarta, contohnya yaitu 20130140133.
- b. Nama_Mhs adalah kolom yang menunjukkan nama dari masing-masing mahasiswa dengan tipe data *Varchar*, pada kolom Nama_Mhs berisi karakter

(*string*) dengan panjang data yang bervariasi tergantung datanya yang mempunyai ukuran panjang data 20 karakter karena merupakan panjang rata-rata nama mahasiswa yang ada di Universitas Muhammadiyah Yogyakarta, contohnya yaitu Sekar Grace Herlia.

- c. Tempat_Lahir adalah kolom yang menunjukkan tempat lahir dari masing-masing mahasiswa dengan tipe data *Varchar*, pada kolom Tempat_Lahir berisi karakter (*string*) dengan panjang bervariasi tergantung datanya yang mempunyai ukuran panjang data 15 karakter karena merupakan panjang rata-rata tempat lahir mahasiswa di Universitas Muhammadiyah Yogyakarta contohnya yaitu Kecapi-Kalianda.
- d. Tgl_Lahir adalah kolom yang menunjukkan tanggal lahir dari masing-masing mahasiswa dengan tipe data *Date* karena pada kolom Tgl_Lahir memiliki format tanggal.
- e. Alamat adalah kolom yang menunjukkan alamat dari masing-masing mahasiswa, dengan tipe data *Text* karena pada kolom Alamat memiliki panjang data yang *relative* lebih panjang dari data di kolom yang lain.
- f. Jenis_Kelamin adalah kolom yang menunjukkan jenis kelamin dari masing-masing mahasiswa dengan tipe data *Enum* karena pada kolom Jenis_Kelamin karakter atau nilai data sudah ditentukan sebelumnya yaitu 'L' atau 'P'.
- g. No_Hp adalah kolom yang menunjukkan nomor *handphone* dari masing-masing mahasiswa dengan tipe data *Varchar* karena No_Hp berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya. Kolom No_Hp

memiliki panjang data 12 karena panjang maksimal dari nomor *handphone* yang ada saat ini adalah 12 karakter.

- h. *Email* adalah kolom yang menunjukkan *email* dari masing-masing mahasiswa, dengan tipe data *Varchar*, *Email* berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 35 karena merupakan panjang rata-rata *email* mahasiswa yang ada di Universitas Muhammadiyah Yogyakarta contohnya yaitu *septi.manjani.2013@ft.umy.ac.id*.
- i. *Id_user* adalah kolom yang menunjukkan *Id_user* dari tabel *users* dan merupakan *foreign key*.

2. Kamus Data Dosen

Tabel 3.2 Kamus Data Tabel Dosen

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
NIDN	Char	10	<i>Primary Key</i>	-
Nama_Dsn	Varchar	35	-	-
Alamat	Text	-	-	-
Jenis_Kelamin	Enum	-	-	-
No_Hp	Varchar	12	-	-
Email	Varchar	35	-	-
Id_user	Int	-	<i>Foreign Key</i>	<i>Users</i>

Penjelasan kamus data dosen pada tabel 3.2 adalah sebagai berikut:

- a. NIDN adalah kolom yang menunjukkan nomor induk dosen nasional dari masing-masing dosen dengan tipe data *Char*, pada kolom NIDN berisi data dengan panjang yang tetap. Kolom NIDN memiliki ukuran panjang data 10 karena merupakan panjang data dari nomor induk dosen nasional, contohnya yaitu 0502026801.
- b. Nama_Dsn adalah kolom yang menunjukkan nama dari masing-masing dosen dengan tipe data *Varchar*, pada kolom Nama_Dsn berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang mempunyai ukuran panjang data 35 karakter karena merupakan panjang rata-rata nama dosen di Universitas Muhammadiyah Yogyakarta, contohnya yaitu Dr. Ir. Dwijoko Purbohadi, M.T.
- c. Alamat adalah kolom yang menunjukkan alamat dari masing-masing dosen, dengan tipe data *Text* karena pada kolom Alamat memiliki panjang data yang *relative* lebih panjang dari data di kolom yang lain.
- d. Jenis_Kelamin adalah kolom yang menunjukkan jenis kelamin dari masing-masing dosen dengan tipe data *Enum* karena pada kolom Jenis_Kelamin karakter atau nilai data sudah ditentukan sebelumnya yaitu 'L' atau 'P'.
- e. No_Hp adalah kolom yang menunjukkan nomor *handphone* dari masing-masing dosen dengan tipe data *Varchar* karena No_Hp berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 12 karena panjang maksimal dari nomor *handphone* yang ada saat ini adalah 12 karakter.

- f. *Email* adalah kolom yang menunjukkan *email* dari masing-masing dosen, dengan tipe data *Varchar*, *Email* berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 35 karakter karena merupakan panjang rata-rata *email* dosen yang ada di Universitas Muhammadiyah Yogyakarta, contohnya yaitu *dpurbohadi@gmail.com*.
- g. *Id_user* adalah kolom yang menunjukkan *Id_user* dari tabel *users* dan merupakan *foreign key*.

3. Kamus Data *Users*

Tabel 3.3 Kamus Data Tabel *Users*

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
<i>Id_user</i>	<i>Int</i>	-	<i>Primary Key</i>	-
<i>Username</i>	<i>Varchar</i>	11	-	-
<i>Password</i>	<i>Varchar</i>	8	-	-
<i>Level</i>	<i>Enum</i>	-	-	-

Penjelasan kamus data *users* pada tabel 3.3 adalah sebagai berikut:

- a. *Id_user* adalah kolom yang menunjukkan *id user* dari masing-masing *user* yaitu mahasiswa atau dosen, bersifat *auto increment* dan *primary key* dengan tipe data *Int* karena pada kolom *Id_user* berisi angka (*numeric*). Kolom *Id_user* memiliki panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.
- b. *Username* adalah kolom yang menunjukkan *username* dari masing-masing *users* yaitu dosen atau mahasiswa dengan tipe data *Varchar*. *Username* yang digunakan adalah NIM dari mahasiswa atau NIDN dari dosen sehingga panjang data *username* 11 karakter.

- c. *Password* adalah kolom yang menunjukkan *password* dari masing-masing *users* yaitu mahasiswa atau dosen dengan tipe data *Varchar* karena *Password* berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 8 karakter untuk mengurangi resiko lupa *password*.
- d. *Level* adalah kolom yang menunjukkan nama dari peran masing-masing *users* yaitu mahasiswa atau dosen dengan tipe data *Enum* karena *Level* diisi dengan salah satu nilai yaitu mahasiswa atau dosen.

4. Kamus Data Matakuliah

Tabel 3.4 Kamus Data Tabel Matakuliah

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
Kode_Matkul	Char	7	Primary Key	-
Nama_Matkul	Varchar	35	-	-
Semester	Enum	-	-	-
Jumlah_SKS	Int	-	-	-
Jumlah_Sesi	Int	-	-	-
Batas_Izin	Int	-	-	-

Penjelasan kamus data matakuliah pada tabel 3.4 adalah sebagai berikut:

- a. *Kode_Matkul* adalah kolom yang menunjukkan kode matakuliah dari masing-masing matakuliah, dengan tipe data *Char* karena pada kolom *Kode_Matkul* berisi data dengan panjang yang tetap. Kolom *Kode_Matkul* memiliki ukuran panjang data 7 karena merupakan panjang data kode matakuliah yang ada di Universitas Muhammadiyah Yogyakarta, contohnya yaitu TI-506P.

- b. Nama_Matkul adalah kolom yang menunjukkan nama dari masing-masing matakuliah dengan tipe data *Varchar* karena pada kolom Nama_ Matkul berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang mempunyai ukuran panjang data 35 karakter karena merupakan panjang rata-rata nama matakuliah yang ada di Universitas Muhammadiyah Yogyakarta, contohnya yaitu Praktikum Pengembangan Aplikasi *Web*.
- c. Semester adalah kolom yang menunjukkan semester dari masing-masing matakuliah dengan tipe data *Enum* karena pada kolom semester karakter atau nilai data sudah ditentukan sebelumnya yaitu '1', '2', '3', '4', '5', '6', '7', atau '8'.
- d. Jumlah_SKS adalah kolom yang menunjukkan jumlah sks dari masing-masing matakuliah, dengan tipe data *Int* karena pada kolom Jumlah_SKS berisi angka (*numeric*) dengan panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.
- e. Jumlah_Sesi adalah kolom yang menunjukkan jumlah pertemuan perkuliahan dari masing-masing matakuliah, dengan tipe data *Int* karena pada kolom Jumlah_Sesi berisi angka (*numeric*) dengan panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.
- f. Batas_Izin adalah kolom yang menunjukkan batas izin perkuliahan dari masing-masing matakuliah, dengan tipe data *Int* karena pada kolom Batas_Izin berisi angka (*numeric*) dengan panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.

5. Kamus Data Perizinan

Tabel 3.5 Kamus Data Tabel Perizinan

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
Id_izin	Int	-	<i>Primary Key</i>	-
NIM	Char	11	<i>Foreign key</i>	Mahasiswa
Kode_Matkul	Char	7	<i>Foreign key</i>	Matakuliah
Tanggal	Date	-	-	-
Alasan	Text	-	-	-
Bukti_Izin	Varchar	30	-	-

Penjelasan kamus data perizinan pada tabel 3.5 adalah sebagai berikut:

- a. Id_izin adalah kolom yang menunjukkan *id* izin dari masing-masing perizinan, bersifat *auto increment* dan *primary key* dengan tipe data *Int* karena pada kolom Id_izin berisi angka (*numeric*) dengan panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.
- b. NIM adalah kolom yang menunjukkan NIM dari tabel mahasiswa dan merupakan *foreign key*.
- c. Kode_Matkul adalah kolom yang menunjukkan Kode_Matkul dari tabel matakuliah dan merupakan *foreign key*.
- d. Tanggal adalah kolom yang menunjukkan tanggal perizinan yang dilakukan dengan tipe data *Date* karena pada kolom Tanggal memiliki format tanggal.

- e. Alasan adalah kolom yang menunjukkan alasan dari masing-masing perizinan, dengan tipe data *Text* karena pada kolom Alasan memiliki panjang data yang *relative* lebih panjang dari data di kolom yang lain.
- f. Bukti_Izin adalah kolom yang menunjukkan bukti izin dari masing-masing perizinan dengan tipe data *Varchar* karena pada kolom Bukti_Izin berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang mempunyai ukuran panjang data 30 karakter contohnya yaitu IMG20170819212308.jpg.

6. Kamus Data Info Tugas

Tabel 3.6 Kamus Data Tabel Info Tugas

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
Id_info	Int	-	<i>Primary Key</i>	-
NIDN	Char	10	<i>Foreign key</i>	Dosen
Kode_Matkul	Char	7	<i>Foreign key</i>	Matakuliah
Tugas	Text	-	-	-
Deadline	Tanggal	-	-	-
Keterangan	Text	-	-	-

Penjelasan kamus data info tugas pada tabel 3.6 adalah sebagai berikut:

- a. Id_info adalah kolom yang menunjukkan *id* info dari masing-masing info tugas, bersifat *auto increment* dan *primary key* dengan tipe data *Int* karena pada kolom Id_info berisi angka (*numeric*) dengan panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.

- b. NIDN adalah kolom yang menunjukkan NIDN dari tabel dosen dan merupakan *foreign key*.
- c. Kode_Matkul adalah kolom yang menunjukkan Kode_Matkul dari tabel matakuliah dan merupakan *foreign key*.
- d. Tugas adalah kolom yang menunjukkan nama dari masing-masing tugas dengan tipe data *Text* karena pada kolom Tugas memiliki panjang data yang *relative* lebih panjang dari data di kolom yang lain.
- e. *Deadline* adalah kolom yang menunjukkan *deadline* tugas yang diberikan dengan tipe data *Date* karena pada kolom *Deadline* memiliki format tanggal.
- f. Keterangan adalah kolom yang menunjukkan keterangan dari masing-masing tugas, dengan tipe data *Text* karena pada kolom Keterangan memiliki panjang data yang *relative* lebih panjang dari data di kolom yang lain.

7. Kamus Data Admin

Tabel 3.7 Kamus Data Tabel Admin

Nama Atribut	Tipe data	Ukuran	Key	Tabel Referensi
Id_admin	Int	-	<i>Primary Key</i>	-
Nama	Varchar	20	-	-
Username	Varchar	15	-	-
Password	Varchar	8	-	-

Penjelasan kamus data admin pada tabel 3.7 adalah sebagai berikut:

- a. Id_admin adalah kolom yang menunjukkan id admin dari masing-masing admin, bersifat *auto increment* dan *primary key* dengan tipe data *Int* karena pada kolom

Id_admin berisi angka (*numeric*). Kolom *Id_admin* memiliki panjang data 11 karena merupakan maksimal panjang data pada tipe data *Int*.

- b. Nama adalah kolom yang menunjukkan nama dari masing-masing admin, dengan tipe data *Varchar* karena pada kolom Nama berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya. Kolom Nama memiliki panjang data 20.
- c. *Username* adalah kolom yang menunjukkan *username* dari masing-masing admin, dengan tipe data *Varchar* karena pada kolom *Username* berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 15 berdasarkan panjang rata-rata *username*.
- d. *Password* adalah kolom yang menunjukkan password dari masing-masing admin, dengan tipe data *Varchar* karena pada kolom *Password* berisi karakter (*string*) dengan panjang data yang bervariasi tergantung datanya yang memiliki panjang data 8 berdasarkan panjang rata-rata *password*.

3.6. Kebijakan Akses Sistem

3.6.1. Kebijakan Akses Mahasiswa

1. *Login*
2. Melihat profil pribadi pada tabel mahasiswa
4. Melakukan perizinan dan melihat data perizinan
5. Melihat data matakuliah
6. Melihat info tugas

3.6.2. Kebijakan Akses Dosen

1. *Login*
2. Melihat, mengubah profil pribadi pada tabel dosen
3. Melihat data perizinan
4. Melihat, menambah, mengubah, menghapus info tugas

3.6.3. Kebijakan Akses Admin

Kebijakan akses admin dilakukan untuk keamanan data, apabila ada seseorang yang dapat membobol *page* admin. Maka kemungkinan data yang didalam *database* masih tetap aman. Berikut adalah kebijakan akses yang hanya bisa dilakukan admin.

1. *Login*
2. Melihat, mengubah profil pribadi pada tabel admin
3. Melihat, menambah, mengubah, menghapus, data mahasiswa
4. Melihat, menambah, mengubah, menghapus, data dosen
5. Melihat, menambah, mengubah, menghapus, data *users*
6. Melihat, menambah, mengubah, menghapus, data matakuliah
7. Melihat, menghapus data izin
8. Melihat, menghapus data info tugas

3.6.4. Kebijakan Akses Master

Master mempunyai hak penuh atas *database*. Master dapat melihat, menambah, mengubah, menghapus data pada *database*.