

Nama : Faizal Wahyu Nuramdhani
NIM : L20018011
Kelas : A

Tugas

1. Rancangan Database menangani data data kuliah

Entities

- Mahasiswa : Menyimpan semua data pribadi mahasiswa
- Dosen : Menyimpan semua data pribadi dosen
- Mata Kuliah : Menyimpan informasi tentang Mata Kuliah
- Ruang_Kelas : Menyimpan informasi tentang transaksi mahasiswa

Attribute

- Mahasiswa :

- NIM_Mahasiswa : Nomor identitas mahasiswa (varchar(10)) PK
- nama_mahasiswa : Nama lengkap mahasiswa (varchar(45))
- alamat_mahasiswa : Alamat lengkap mahasiswa (varchar(100))

- Dosen :

- nip_dosen : nomor induk pegawai (integer) PK
- nama_dosen : nama lengkap dosen (varchar(45))
- alamat_dosen : alamat lengkap dosen (varchar(100))

- Mata_Kuliah :

- kode_kuliah : kode identitas mata kuliah (varchar(10)) PK
- nama_matkul : nama mata kuliah (varchar(45))
- jumlah_sks : jumlah sks per 1 mata kuliah (integer)

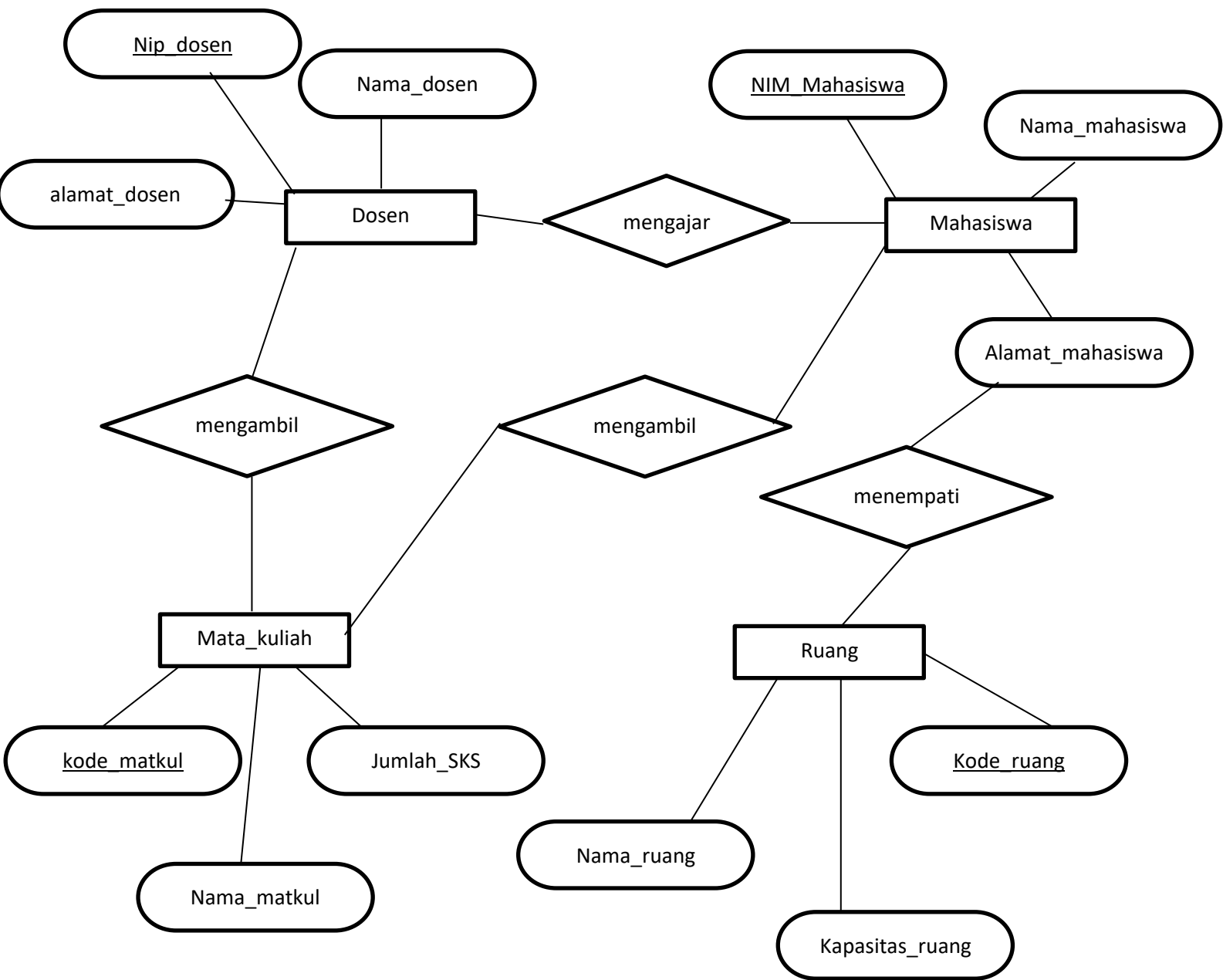
- Ruang_Kelas :

- kode_ruang : kode identitas ruangan (integer) PK
- nama_ruang : nama ruangan (varchar(50))
- kapasitas_ruang : jumlah maksimal mahasiswa dalam 1 ruang (integer)

	Mahasiswa	Dosen	Matkul	Ruang
Mahasiswa	-	n:n	n:n	1:1
Dosen		-	n:n	-
Matkul			-	-
Ruang				-

Hubungan

- **Dosen mengajar Mahasiswa**
 - Tabel utama : **mahasiswa, dosen**
 - Tabel kedua : **mahasiswa_has_dosen**
 - Relationship : Many-to-Many (n:n)
 - Attribute_Penghubung : **NIM_mahasiswa, nip_dosen** (FK NIM_mahasiswa, nip_dosen di mahasiswa_has_dosen)
- **Mahasiswa mengambil mata_kuliah**
 - Tabel utama : **mahasiswa, mata_kuliah**
 - Tabel kedua : **mengambil**
 - Relationship : Many-to-Many (n:n)
 - Attribute_Penghubung : **NIM_mahasiswa, kode_matkul** (FK NIM_mahasiswa, kode_matkul di mengambil)
- **Mahasiswa menempati ruang_kelas**
 - Tabel utama : **mahasiswa**
 - Tabel kedua : **ruang_kelas**
 - Relationship : One-to-One (1:1)
 - Attribute_Penghubung : **NIM_mahasiswa** (FK NIM_mahasiswa di ruang_kelas)
- **Dosen mengambil mata_kuliah**
 - Tabel utama : **dosen, mata_kuliah**
 - Tabel kedua : **dosen_has_matakuliah**
 - Relationship : Many-to-Many (n:n)
 - Attribute_Penghubung : **nip_dosen, kode_matkul** (FK nip_dosen, kode_matkul di - dosen_has_matakuliah)



2. Contoh sembarang database

Penjualan di apotek

Entities

- Pelanggan : menyimpan semua data pelanggan
- pegawai : menyimpan semua data pegawai
- obat : menyimpan informasi obat

Attributes

- *Pelanggan:*

- id_pelanggan : nomor identitas pelanggan (integer)PK
- nama_pelanggan : nama lengkap pelanggan (varchar(45))
- alamat_pelanggan: alamat lengkap pelanggan (varchar(100))

- *Pegawai :*

- id_pegawai : nomor identitas pegawai (integer)PK
- nama_pegawai : nama lengkap pegawai (varchar(45))
- alamat_pegawai : alamat lengkap pegawai (varchar(100))

- *obat :*

- Kode_obat : kode obat (varchar(10)) PK
- nama_obat : nama obat (varchar(45))
- harga_obat : harga setiap obat (integer)

	Pelanggan	Pegawai i	obat
Pelanggan	-	n:n	n:n
Pegawai		-	1:n

Obat			-
-------------	--	--	---

Hubungan

- **Pegawai melayani pelanggan**
 - Tabel utama : **pegawai, pelanggan**
 - Tabel kedua : **pelanggan_has_pegawai**
 - Relationship : Many-to-many (n:n)
 - Attribute penghubung : **id_pegawai, id_pelanggan** (FK id_pegawai, id_pelanggan di **pelanggan_has_pegawai**)
- **Pelanggan membeli obat**
 - Tabel utama : **pelanggan, obat**
 - Tabel kedua : **membeli**
 - Relationship : Many-to-many(n:n)
 - Attribute penghubung : **id_pelanggan, kode_obat**(FK id_pelanggan, kode_obat di **membeli**)
- **Pegawai memberikan Obat**
 - Tabel utama : **pegawai, obat**
 - Tabel kedua : **memberikan**
 - Relationship : One-to-many(1:n)
 - Attribute penghubung : **id_pegawai, kode_obat**(FK id_pegawai, kode_obat di **memberikan**)

