

PERTEMUAN 11

PERANCANGAN *ENTITY RELATIONSHIP* DIAGRAM

A. TUJUAN PEMBELAJARAN

Mahasiswa dapat menganalisa bagaimana setiap entitas berelasi dari suatu kasus yang diberikan, dan kemudian membangun modelnya dengan menggunakan ERD.

B. URAIAN MATERI

1. Pemetaan Kardinalitas

Sejak ER-D dibuat, setiap entitas yang ditautkan ke sebuah entitas lain harus dibatasi pada jumlah/atribut minimum dan maksimum dari kemunculan entitas. Peristiwa ini dapat menunjukkan kemungkinan adanya hubungan antar entitas.

Sejumlah kemungkinan hubungan antar entitas digunakan untuk menentukan batas minimum dan maksimum yang dapat atau harus terjadi dari satu entitas ke entitas lain dan sebaliknya.

Menampilkan angka minimum dan maksimum yang dapat dihubungkan di setiap entitas disebut kardinalitas. Yang dimaksud dengan kardinalitas adalah:

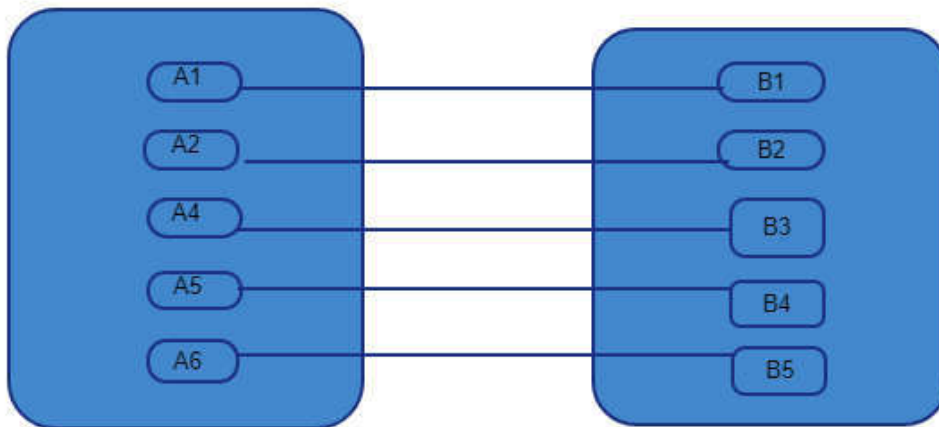
Mendefinisikan batas minimum dan maksimum untuk setiap peristiwa dari suatu entitasnya yang dapat dihubungkan ke suatu peristiwa di entitas lain. Karena semua relasi bersifat langsung, kardinal harus didefinisikan secara langsung di antara dua entitas kapan pun mereka terhubung (whitten et.al.,2001:264) Atau dapat pula dikatakan sebagai berikut:

Analisis terhadap nilai minimal dan maksimal yang mungkin terjadi pada suatu hubungan antara sebuah entitas dengan entitas lainnya.

Dalam ER-D hubungan antara entitas dapat dipetakan menjadi beberapa pembatas yaitu :

a. One to one

Setiap entitas dapat berhubungan dengan paling banyak dengan satu entitas lain, dan sebaliknya. Misalnya, sebuah entitas A ditugaskan ke paling banyak satu entitas B. Ini berarti bahwa entitas A hanya terhubung ke maksimum satu entitas B. Lebih jelasnya dapat dilihat pada gambar berikut:

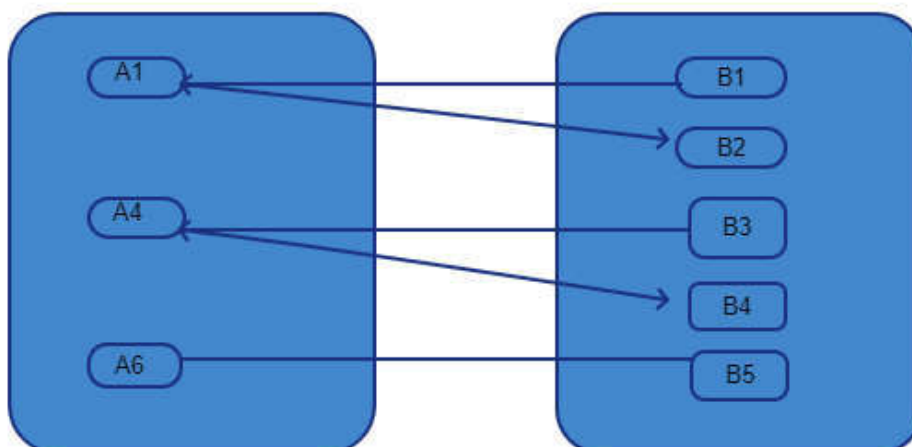


Gambar 0.1 Hubungan One to one (1 – 1)

b. Satu ke banyak (1-M / N)

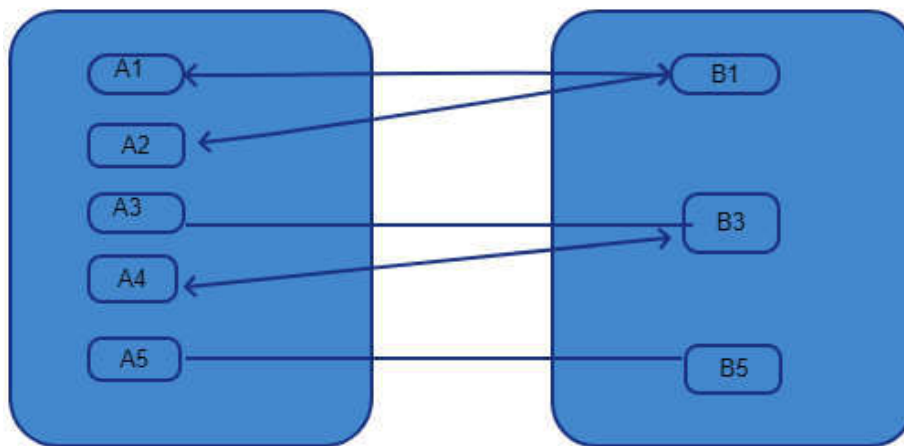
Disebut juga pemetaan satu ke banyak yaitu bahwa suatu atribut dapat merujuk ke lebih dari satu atau banyak atribut lain, tapi tidak dapat dilakukan sebaliknya.

Misalnya, sebuah entitas A terhubung ke lebih dari satu ke entitas B. Tetapi banyak atribut B hanya satu hubungan atribut A. seperti contoh gambar berikut.



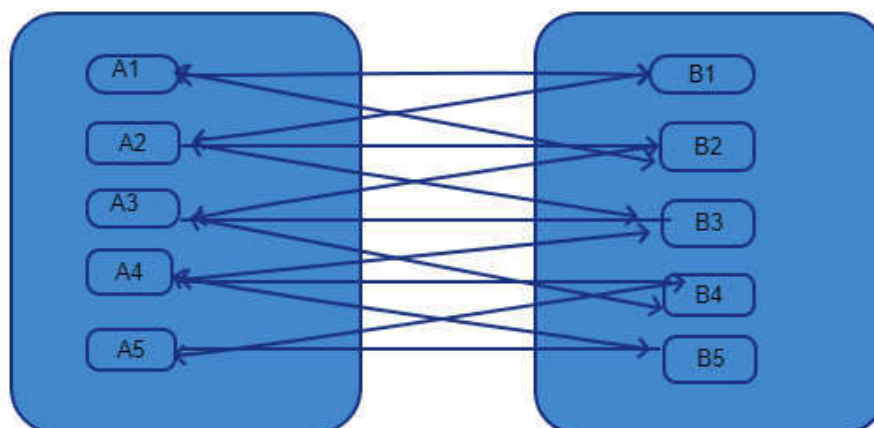
Gambar 0.2 many-to-one (M / N-1)

Hubungan Many to One adalah kebalikan dari sebuah hubungan *One to many*. Harus lebih dari satu entitas hanya berhubungan dengan salah satu entitas lain, tetapi tidak dapat dilakukan sebaliknya. Misalnya, sebuah entitas A hanya berhubungan ke satu entitas B, tetapi entitas B bisa dihubungkan ke banyak entitas A. Contohnya dapat dilihat pada gambar di bawah ini..

**Gambar 0.3** hubungan many to one (M / N-1)

c. *Many to many* (M-N)

Banyak entitas dapat terhubung ke banyak entitas yang lain, dan sebaliknya. Misalnya, banyak entitas A terhubung ke entitas B dan sebaliknya, entitas B berhubungan dengan banyak entitas A. Contohnya dapat dilihat pada gambar di bawah ini.

**Gambar 0.4** hubungan many-to-many (M-N)

2. Perancangan ERD

Pembuatan ER-D selalu dilakukan dengan bertahap biasanya menggunakan dua tahap, yaitu:

a. Tahap awal pembuatan diagram (preliminary design)

Tahapan awal bertujuan untuk dapat sebuah gambaran basis data yang minimal dapat mengakomodasikan kebutuhan penyimpanan data untuk semua kegiatan sistem atau memulai usaha. Pembuatan desain awal ini hanya bertujuan untuk membuat gambaran secara garis besar dan dapat memperlihatkan adanya hubungan antar entitas dan belum dilihatkan relasi kardinalnya. Tingkat tersebut disebut juga dengan *enterprise level*.

b. Tahapan optimasi diagram E-R (final design)

Tahap optimasi pada ER terbagi menjadi 2 (dua) yaitu:

a) *Entity relationship level*

Tingkatan ini hanya memperlihatkan adanya hubungan antar entitas, namun belum dibuat atribut pada tiap-tiap entitas dan belum dilihatkan relasi kardinalnya. pembuatan ini bertujuan untuk memudahkan bila terjadi perubahan terhadap penambahan entitas atau perubahan relasi

b) *Entity relationship attribute level*

Tingkatan ini telah terbentuk diagram ER dengan lengkap (terdapat atribut). Pembuatan tahap ini biasanya dilakukan bila adanya keyakinan analis sistem bahwa diagram ER yang dibuat telah sempurna atau final (tidak ada perubahan lagi).

Pada tahap awal pembuatan diagram awal ada tatacara dan aturan dalam pembuatannya antara lain sebagai berikut:

(1) Menentukan Entitas

Pada tahap ini kita harus mencari entitas-entitas yang terkait dengan sistem yang akan dibuat.

(2) Menentukan atribut key dari setiap entitas

Setelah kita mendapatkan entitas yang dibutuhkan tahap selanjutnya adalah menentukan atribut kunci dari masing-masing entitas. Tujuan ditentukannya atribut key atau kunci adalah untuk menghubungkan antar entitas tersebut.

(3) Menentukan relasi

Pada tahap ini setelah menemukan entitas-entitas terkait selanjutnya adalah menentukan relasi antar entitas. Baik itu one to one, *one to many*, atau *many to many*.

(4) Menggambar diagram awal (preliminary design)

Bila tahap menentukan entitas dan relasi telah selesai kita dapat langsung melakukan perancangan *database* tahap awal dengan cara:

Buat persegi Panjang untuk semua entitas dan beri nama dengan benar. Identifikasikan hubungan antar entitas dan hubungkan mereka dengan menggunakan garis dan tambahkan berlian ditengah gambar hubungan mereka yang telah menggambarkan hubungan. Tambahkan atribut untuk entitas dan beri nama atribut tersebut agar mudah dipahami.

Bila telah selesai melakukan semua tahap tersebut artinya kita telah menyelesaikan tahap *entity relationship level* dimana tahap ini memperlihatkan adanya hubungan antar entitas, namun belum dibuat atribut pada tiap-tiap entitas dan belum dilihatkan relasi kardinalnya. Dan pada tahap *entity relationship attribute level* diharuskan mengisi kardinalitas dan atribut yang bergantung pada entitas.

3. Contoh Kasus dalam Pembuatan ERD

Contoh kasus 1 :

Sebuah lembaga pendidikan Bahasa “LPB Semoga Lancar” mempunyai 10 kantor cabang di area JABODETABEK. Materi kursus yang ditawarkan pada setiap cabang adalah Bahasa Inggris, Jerman, Perancis, Belanda, Mandarin, Jepang. Setiap materi kursus memiliki biaya yang berbeda-beda untuk waktu lama waktu belajar yang bervariasi pula.

Pada setiap cabang, setiap materi kursus dapat diselenggarakan dalam beberapa kelas sesuai dengan jumlah peserta yang mendaftar untuk materi tersebut. Sedangkan lama waktu kursus diselenggarakan dihitung dalam term (kuartal), dimana seorang peserta dalam satu term hanya boleh ikut satu materi saja di salah satu cabang.

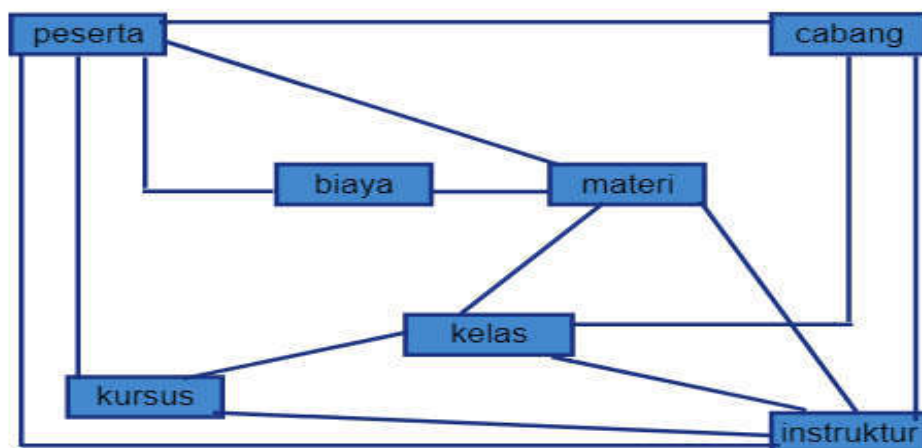
Seorang instruktur pada lembaga tersebut hanya diizinkan mengajar satu materi kursus, tetapi dapat mengajar materi tersebut pada beberapa kelas disemua cabang.

Berdasarkan uraian diatas gambarkan diagram E-R dari basis data yang merupakan model data pada lembaga tersebut. Untuk setiap entitas anda di bebaskan untuk memberi atribut yang sesuai dengan entitas tersebut.

Pemecahan kasus :

- a. Pada tahap awal tentukan entitas entitas mana yang berhubungan dengan sistem yang akan dibuat setelah itu hubungkan antar entitas tersebut. bertujuan agar mendapatkan gambaran basis data minimal dapat mengakomodasikan kebutuhan penyimpanan data untuk semua kegiatan sistem.

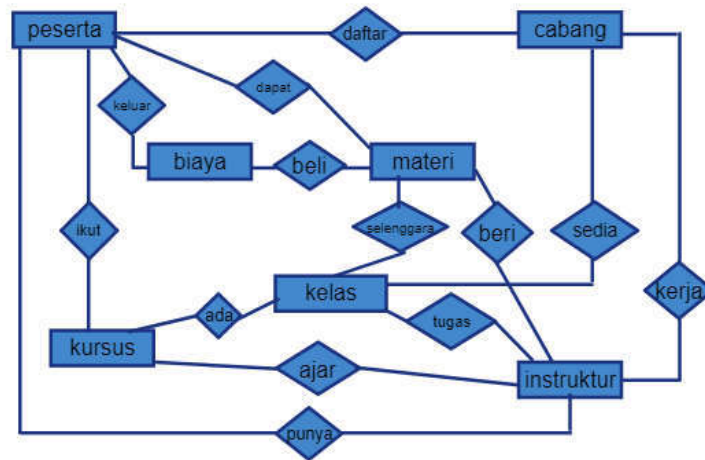
Berikut ini adalah hasil sebuah rancangan awal.



Gambar 0.5 awal kasus 1

- b. Bila Rancangan awal telah terbentuk tahapan selanjutnya adalah membuat relasi/hubungan dari antar entitas tersebut. Pada tahap ini disebut dengan *Entity Relationship Level*.

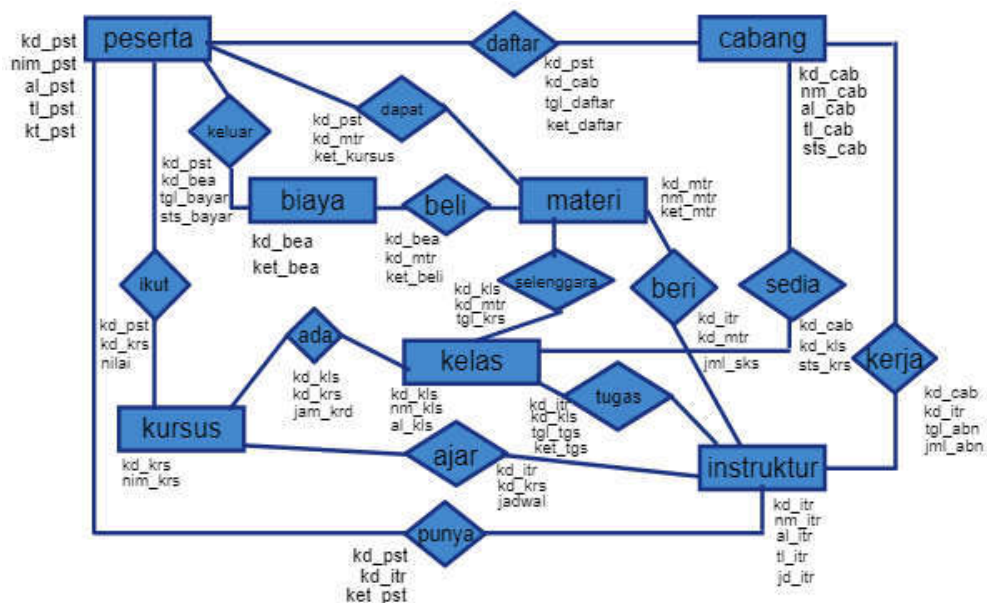
Berikut hasil dari *entity relationship level* yang telah dibuat.



Gambar 0.6 Entity Relationship Level kasus 1

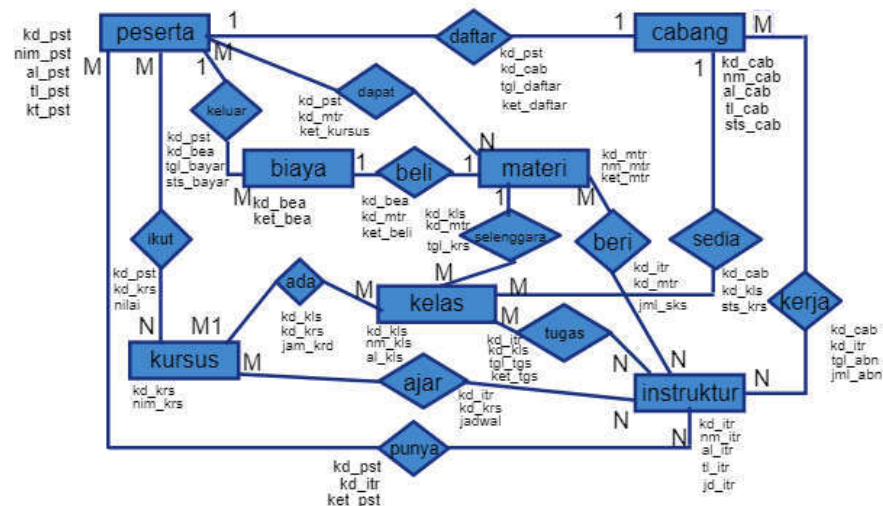
- c. Setelah tahap *Entity Relationship Level* terbentuk maka lengkapilah atribut-atribut yang menyertai setiap entitas, Pada tahap ini disebut juga dengan *entity relationship attribute level*.

Berikut hasil dari *entity relationship attribute level* yang telah dibuat.



Gambar 0.7 Entity Relationship Attribute Level kasus 1

- d. Setelah tahap *Entity Relationship Attribute Level* terbentuk maka lengkapilah Degan mengisi kardinalitas antar entitasnya.



Gambar 0.8 kardinalitas ER-D kasus 1

Kamus Data :

- Peserta {kd_pst, nm_pst, al_pst, tl_pst, kt_pst}
- Cabang {kd_cab, nm_cab, al_cab, tl_cab, sts_cab}
- Biaya {kd_bea, ket_bea}
- Materi {kd_materi, nm_mtr, ket_mtr,}
- Kelas {kd_ks, nm_ks, al_ks}
- Kursus {kd_krs, nm_krs}
- Intrusktur {kd_itr, nm_itr, al_itr, tl_itr, jd_itr}
- Daftar {kd_pst, kd_cab, tgl_daftar, ket_daftar}
- Dapat {kd_pst, kd_mtr, ket_kursus}
- Keluar {kd_pst, kd_bea, tgl_bayar, sts_bayar}
- Beli {kd_bea, kd_mtr, ket_beli}
- Sedia{kd_cab, kd_mtr, tgl_mtr, sts_mtr}
- Kerja {kd_cab, kd_itr, tgl_abn, jml_abn}
- Ikut {kd_pst, kd_krs, nilai}

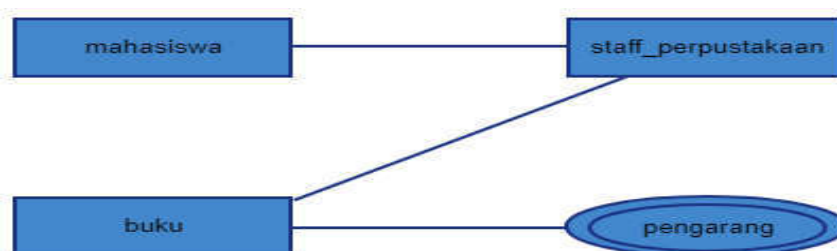
- Ada {kd_kls, kd_krs, nilai}
- Selenggara {kd_kls, kd_mtr, tgl_krs}
- Beri {kd_cab, kd_kls, sts_krs}
- Kasih {kd_itr, kd_mtr, jml_sks}
- Tugas {kd_itr, kd_kls, tgl_kls, ket_tgs}
- Ajar {kd_itr, kd_krs, jadwal}
- Punya {kd_pst, kd_itr, ket_pst}

Contoh kasus 2 :

Seorang mahasiswa ingin mendaftar menjadi anggota peserta perpustakaan falkultas disebuah kampus agar bisa meminjam buku disana, maka sebagai syarat untuk mendaftar dicatatlah nomer induk mahasiswa, nama mahasiswa, alamat mahasiswa, nomer telepon. Setelah itu baru mahasiswa tersebut meminjam buku ke perpustakaan tersebut. Buku-buku yang tersedia diperpustakaan tersebut jumlahnya banyak sekali dan setiap bukunya memiliki data judul, pengarang, nomor buku. Berdasarkan uraian diatas maka gambarkan diagram E-R dari basis data yang merupakan model data tersebut. Serta tentukan pula entitas atribut dan relasinya

Pemecahan kasus :

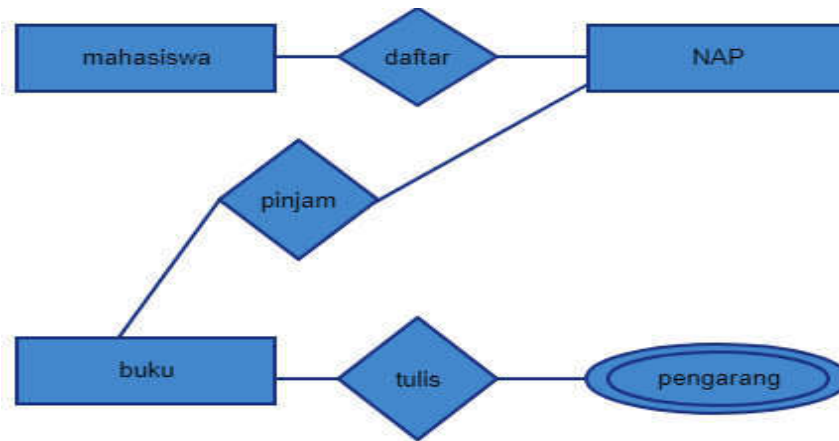
- a. Seperti pada contoh kasus yang pertama buatlah rancangan awal yaitu tentukan entitas entitas mana yang berhubungan dengan sistem yang akan dibuat setelah itu hubungkan antar entitas tersebut bertujuan untuk dapat sebuah gambaran basis data yang minimal yang mengakomodasikan kebutuhan sebuah penyimpanan data ke semua kegiatan *system* yang ada. Berikut adalah hasil rancangan awal nya



Gambar 0.9 rancangan awal kasus 2

- b. Setelah rancangan awal telah dibuat lalu ketahap selanjutnya yaitu membuat relasi/hubungan dari dari antar entitas tersebut atau *Entity Relationship Level*.

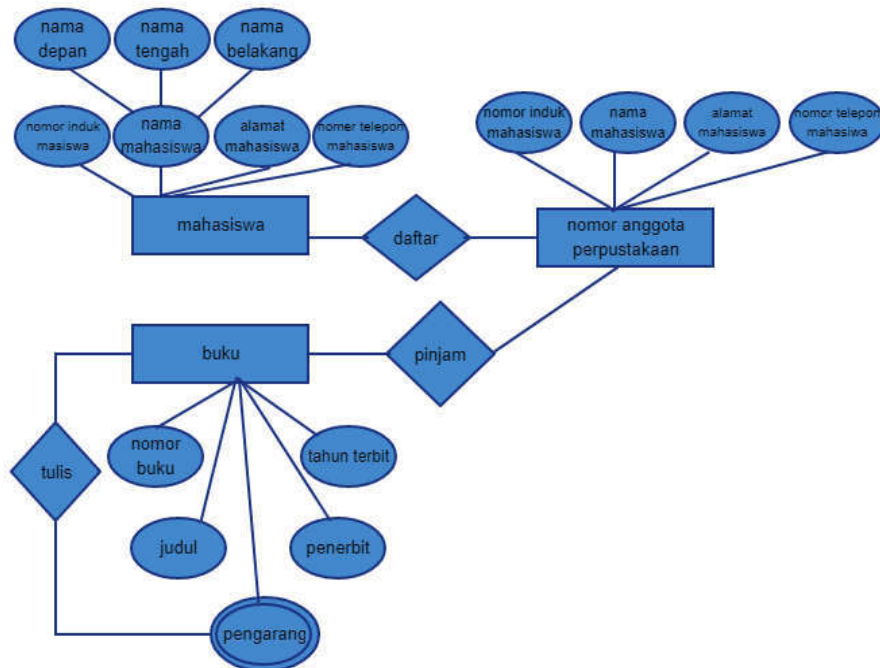
Berikut hasil dari *Entity Relationship Level*.



Gambar 0.10 *Entity Relationship Level* kasus 2

- c. Setelah tahap *Entity Relationship Level* terbentuk maka lengkapilah atribut-atribut yang menyertai setiap entitas, Pada tahap ini disebut juga dengan *entity relationship attribute level*.

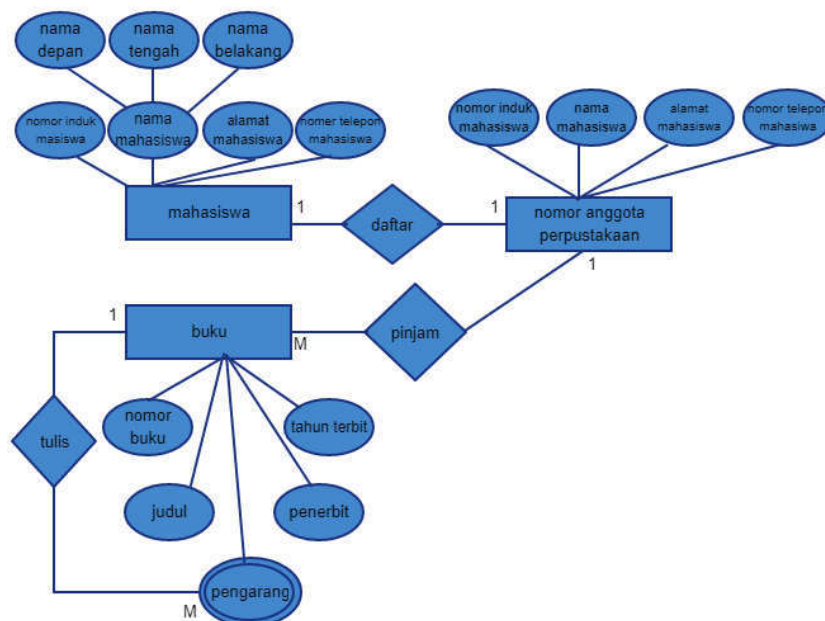
Berikut hasil dari *entity relationship attribute level* yang telah dibuat



Gambar 0.11 *Entity Relationship Attribute Level* kasus 2

- d. Setelah *Entity Relationship Attribute Level* terbentuk maka lengkapilah Degan mengisi kardinalitas antar entitasnya.

Berikut adalah hasil kardinalitas nya.



Gambar 0.12 kardinalitas ER-D kasus 2

Berikut adalah daftar entitas atribut dan relasi / kardinalitas nya:

- a) Entitas: mahasiswa, nomer anggota mahasiswa, buku.
- b) Atribut: nomor induk mahasiswa, nama mahasiswa, nama depan mahasiswa, nama tengah mahasiswa, nama belakang mahasiswa, alamat mahasiswa, nomer telepon, nomor buku, pengarang, penerbit, judul, tahun terbit.
- c) Relasi: daftar, pinjam.

C. SOAL LATIHAN/TUGAS

1. Jelaskan pengertian kardinalitas !
2. Jelaskan dan berikan contoh dari kardinalitas one to one !
3. Jelaskan dan berikan contoh dari kardinalitas *one to many* !
4. Jelaskan dan berikan contoh dari kardinalitas *many to many* !
5. Sebutkan tahapan-tahapan dalam perancangan ER-D !
6. Jelaskan apa yang dimaksud dengan pembuatan diagram awal (preliminary design)
7. Jelaskan apa yang dimaksud dengan pembuatan Tahapan optimasi diagram E-R (final design).

D. REFERENSI

- Adyanata Lubis, S.Kom., M.Kom (2016), Basis Data Dasar Untuk Mahasiswa Ilmu Komputer, Pasir pengairan, CV BUDI UTAMA.
- Fatansyah (2002), Buku Teks Ilmu Komputer : Basis data, Bandung, Informatika.
- Coronel, C., & Morris, S. (2017). *Database System: Design, Implementation, & Management, 13th Edition*. Boston: Cengage Learning, Inc.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011). *Database System Concept ; Sixth Edition*. New York: McGraw-Hill

GLOSARIUM

One-to-one adalah satu-ke-satu dalam ER-D berarti bahwa entitas akan berhubungan paling banyak dengan masing-masing satu entitas lainnya .

One-to-many adalah satu atribut yang dapat berhubungan dengan lebih banyak atribut yang lain tetapi tidak sebaliknya.

Many-to-Many adalah banyak entitas yang dapat dihubungkan dengan banyak entitas-entitas lainnya.