

PERKULIAHAN KE 5

ENTITY RELATIONSHIP DIAGRAM (ERD)

4.1 Pengertian *Entity Relationship Diagram* (ERD)

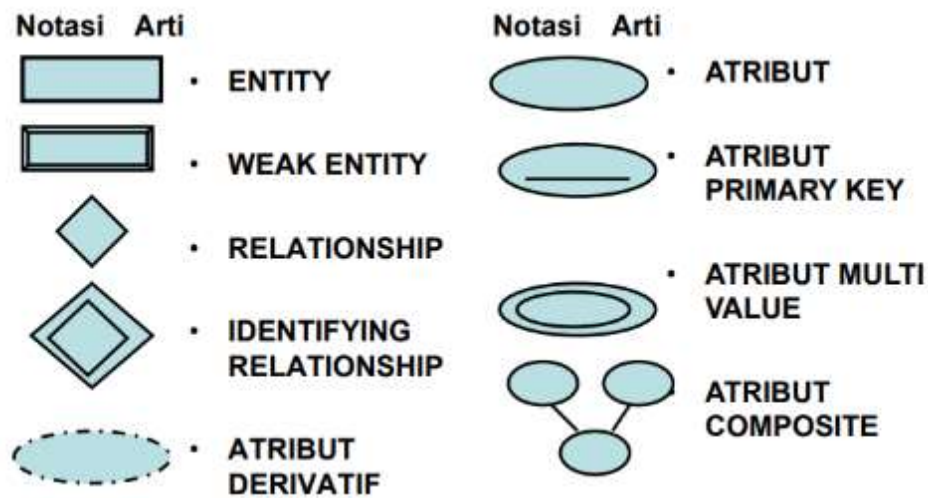
Beberapa pengertian mengenai ERD oleh beberapa ahli:

1. Menurut (Ladjamudin, 2004), ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.
2. Menurut (Fathansyah, 2012), ERD adalah model entity relationship yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta ‘dunia nyata’ yang kita tinjau.
3. Menurut (Sukamto & Shalahuddin, 2018), *Entity Relationship Diagram* (ERD) adalah bentuk paling awal dalam melakukan perancangan basis data relasional. jika menggunakan OODBMS maka perancangan ERD tidak diperlukan.

Biasanya ERD ini digunakan oleh professional sistem untuk berkomunikasi dengan pemakai eksekutif tingkat tinggi dalam suatu organisasi. Pemakai ini lebih tertarik dengan hal-hal sebagai berikut :

- a. Data apa saja yang dibutuhkan untuk bisnis mereka?
- b. Bagaimana data tersebut berelasi dengan data lainnya?
- c. Siapa saja yang diperkenankan untuk mengakses data tersebut?

4.2 Komponen ERD



Gambar IV.1
Komponen ERD

1. Entitas

Entitas adalah suatu kumpulan object atau sesuatu yang dapat dibedakan atau dapat diidentifikasi secara unik. Dan kumpulan entitas yang sejenis disebut dengan entity set. Entity Set terbagi menjadi dua, yaitu :

a. *Strong entity set*

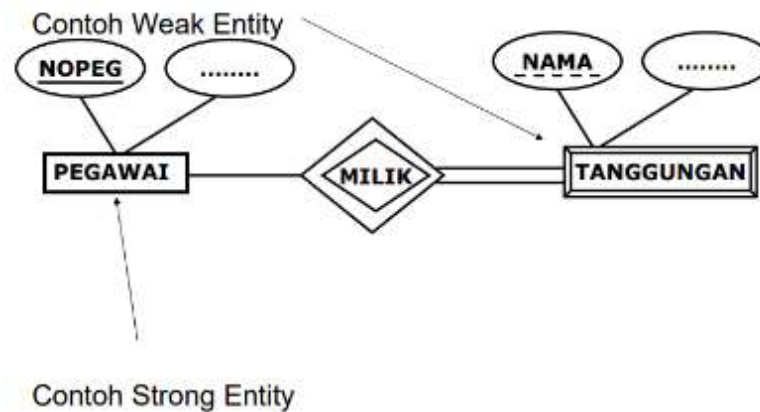
Yaitu entity set yang satu atau lebih atributnya digunakan oleh entity set lain sebagai key. Digambarkan dengan empat persegi panjang. Misal : E adalah sebuah entity set dengan attribute-atribute a_1, a_2, \dots, a_n , maka entity set tersebut direpresentasikan dalam bentuk tabel E yang terdiri dari n kolom, dimana setiap kolom berkaitan dengan attribute-atributnya.

b. *Weak Entity set*

Yaitu Entity set yang bergantung terhadap *strong entity set*. Digambarkan dengan empat persegi panjang bertumpuk. Misal : A adalah *weak entity set*

dari attribute-attribute a_1, a_2, \dots, a_r dan B adalah strong entity set dengan attribute-attribute b_1, b_2, \dots, b_s , dimana b_1 adalah attribute primary key, maka weak entity set direpresentasikan berupa table A, dengan attribute-attribute $\{b_1\} \cup \{a_1, a_2, \dots, a_r\}$

Contoh :



Gambar IV.2
Contoh Strong dan Weak Entity Set

2. Atribut

Atribut adalah kumpulan elemen data yang membentuk suatu entitas.

Jenis-jenis atribut :

a. Atribut Kunci

Merupakan atribut yang digunakan untuk menentukan suatu entity secara unik

b. Atribut simple

Merupakan atribut yang bernilai tunggal

c. Atribut Multi Value

Merupakan atribut yang memiliki sekelompok nilai untuk setiap instan entity.

Contoh : Pada gambar dibawah ini, yang menjadi atribut key adalah NIP. Tgl Lahir dan Nama adalah atribut simple. Sedangkan Gelar merupakan contoh atribut multivalue.



Gambar IV.3
Contorh Atribut Key, Simple dan Multivale

d. Atribut Composit

Merupakan suatu atribut yang terdiri dari beberapa atribut yang lebih kecil yang mempunyai arti tertentu. Contohnya adalah atribut nama pegawai yang terdiri dari nama depan, nama tengah dan nama belakang.

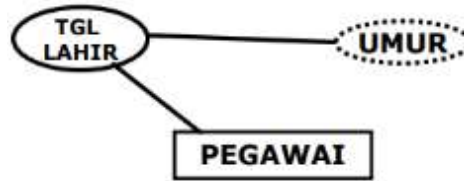


Gambar IV.4
Contoh Atribut Composit

e. Atribut Derivatif

Merupakan suatu atribut yg dihasilkan dari atribut yang lain. Sehingga umur yang merupakan hasil kalkulasi antara Tgl Lahir dan tanggal hari ini.

Sehingga keberadaan atribut umur bergantung pada keberadaan atribut Tgl Lahir.



Gambar IV.5
Contoh Atribut Derivatif

3. *Relationship*

Relationship merupakan hubungan yang terjadi antara satu entitas atau lebih.

Participation Constraint menjelaskan apakah keberadaan suatu entity tergantung pada hubungannya dengan entity lain. Terdapat dua macam *participation constrain* yaitu:

a. *Total participation constrain*

Yaitu Keberadaan suatu entity tergantung pada hubungannya dengan entity lain. Didalam diagram ER digambarkan dengan dua garis penghubung antar entity dan relationship.

b. *Partial participation*

Yaitu Keberadaan suatu entity tidak tergantung pada hubungan dengan entity lain. Didalam diagram ER digambarkan dengan satu garis penghubung.

Contoh :

a. TOTAL PARTICIPATION



b. PARTIAL PARTICIPATION



Gambar IV.6
Contoh Jenis Participation Constrains

4. *Indicator Type*a. *Indicator Asosiatif Object*

Indicator tipe asosiatif object berfungsi sebagai suatu objek dan suatu relationship. Contoh :

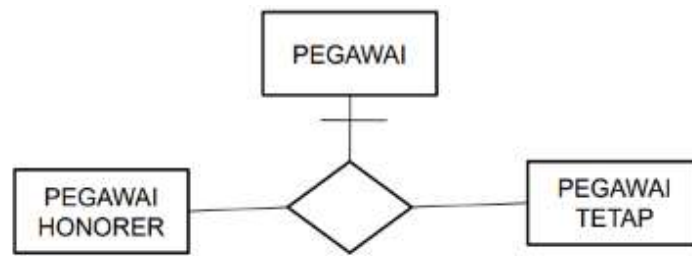


Gambar IV.7
Contoh *Indicator Asosiatif Object*

b. *Indicator Super Tipe*

Indicator tipe super tipe, terdiri dari suatu object dan satu subkategori atau lebih yang dihubungkan dengan satu relationship yang tidak bernama.

Contoh :



Gambar IV.7
Contoh *Indicator Supert type*

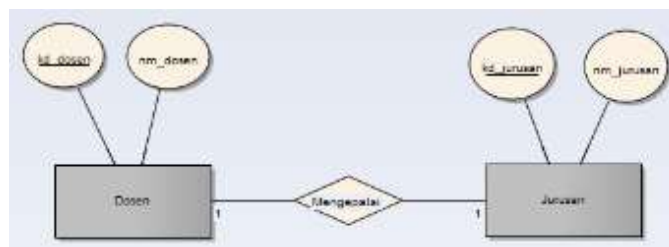
4.3 Kardinalitas / Derajat Relasi

Menurut (Fathansyah, 2012), Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Jenis-jenis kardinalitas sebagai berikut:

1. *One to One* (1:1)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

Contoh :



Gambar IV.8
Contoh *One to One*

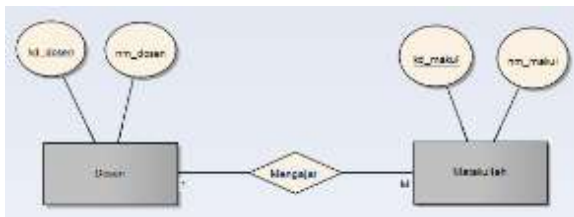
Pada relasi ini, setiap dosen paling banyak mengepali satu jurusan (walaupun memang tidak semua dosen yang menjadi ketua jurusan). Dan setiap jurusan pasti dikepalai oleh paling banyak satu orang dosen.

2. *One to Many* (1:M) atau sebaliknya *Many to One* (M:1)

One to Many berarti setiap entitas pada himpunan entitas A berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

Many to One berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan banyak entitas pada himpunan entitas A.

Contoh :



Gambar IV.9
Contoh *One to Many*

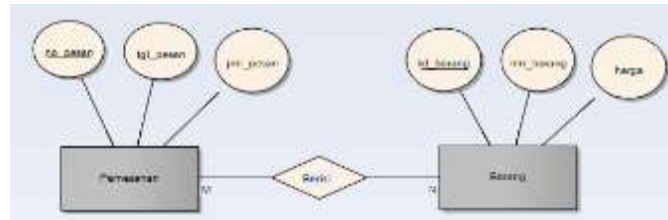
Pada relasi ini, setiap dosen dapat mengajar lebih dari satu matakuliah, sedang setiap matakuliah diajar hanya oleh paling banyak satu orang dosen.

3. *Many to Many* (M:N)

Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian sebaliknya, dimana

setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

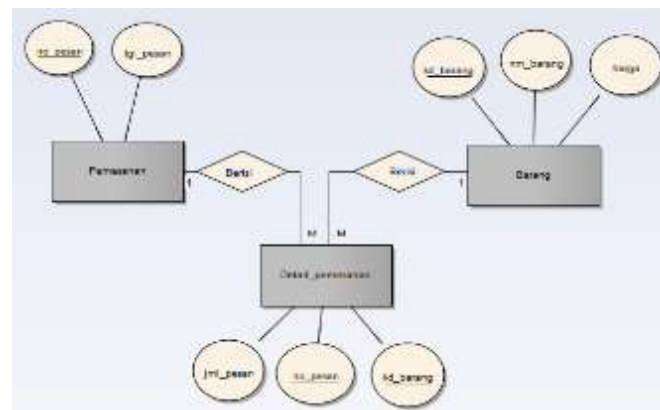
Contoh :



Gambar IV.10
Contoh *Many to Many*

Pada relasi diatas setiap pemesanan bias berisi banyak barang dan setiap barang mengisi banyak pemesanan. Jika kardinalitasnya Many to Many ini dapat disederhanakan kembali, karena munculnya entitas yang baru.

Penggambarannya sebagai berikut :



Gambar IV.11
Contoh *Many to Many*

Pada ERD yang baru ini, muncul entitas yang baru yaitu Detail_pemesanan. Dan dalam entitas Detail_pemesanan, terdapat no_pesan dan kd_barang sebagai *Foreign Key*.

4.4 Tahapan Pembuatan ERD

Menurut (Hidayatullah & Kawistara, 2015), Tahpan pembuatan ERD sebagai berikut :

1. Identifikasi dan tetapkan seluruh himpunan entitas yang akan terlibat
2. Tentukan atribut key dari masing-masing himpunan entitas
3. Identifikasi dan tetapkan seluruh himpunan relasi antar himpunan entitas yang ada beserta foreign key-nya
4. Tentukan derajat/kardinalitas relasi untuk setiap himpunan relasi
5. Lengkapi himpunan entitas dan himpunan relasi dengan atribut bukan kunci.

4.5 *Logical Record Structured (LRS)*

Merupakan representasi dari struktur record-record pada tabel-tabel yang terbentuk dari hasil relasi antar himpunan entitas.

Menentukan Kardinalitas, Jumlah Tabel dan Foreign Key (FK)

1. *One To One*

Contoh :

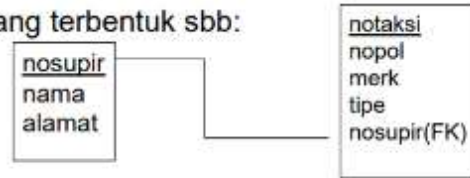


Gambar IV.12
Contoh *One To One*

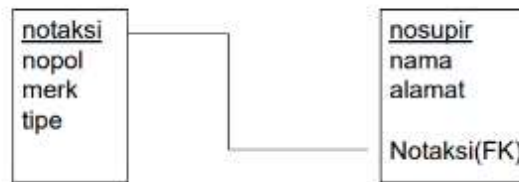
Gambar di atas menunjukan relasi dengan kardinalitas 1-1, karena: 1 supir hanya bisa mengemudikan 1 taksi, dan 1 taksi hanya bisa dikemudikan oleh 1 supir. Relasi 1-1 akan membentuk 2 tabel:

- a. Tabel Supir (nosupir, nama, alamat)
- b. Tabel Taksi (notaksi, nopol, merk, tipe)

LRS yang terbentuk sbb:



atau



Gambar IV.13
Contoh LRS

2. *One To Many*

Contoh :

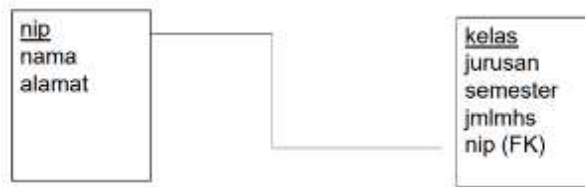


Gambar IV.14
Contoh *One to Many*

Gambar di atas menunjukkan relasi dengan kardinalitas 1-M, karena: 1 Dosen bisa membimbing banyak Kelas, dan 1 Kelas hanya dibimbing oleh 1 Dosen.

Relasi 1-M akan membentuk 2 tabel:

- a. Tabel Dosen (nip, nama, alamat)
- b. Tabel Kelas (kelas, jurusan, semester, jmlmhs)



Gambar IV.15
Contoh *LRS*

3. *Many To Many*

Contoh :



Gambar IV.16
Contoh *Many to Many*

Gambar di atas menunjukkan relasi dengan kardinalitas M-M, karena: 1 Mahasiswa bisa belajar banyak Mata Kuliah, dan 1 Mata Kuliah bisa dipelajari oleh banyak Mahasiswa. Relasi M-M akan membentuk 3 tabel:

- Tabel Mahasiswa (nim, nama, alamat)
- Tabel Mtkuliah (kdmk, nmmk, sks)
- Tabel Nilai (nim, kdmk, nilai) → menggunakan super key/composite key



Gambar IV.17
Contoh *LRS*

4.6 Membuat ERD

Langkah-langkah pembuatan ERD dan LRS:

1. Tentukan entity-entity yang diperlukan
2. Tentukan relationship antar entity-entity
3. Menggambar ERD Sementara
4. Mengisi kardinalitas
5. Menentukan kunci utama
6. Menggambar ERD Berdasarkan Kunci
7. Tentukan attribute-attribute
8. Transformasi ERD ke LRS Menggambar LRS

Contoh :

Sebuah perusahaan mempunyai beberapa bagian. Masing-masing bagian mempunyai pengawas dan setidaknya satu pegawai. Pegawai harus ditugaskan pada paling tidak satu bagian, tetapi dapat pula beberapa bagian. Paling tidak satu pegawai mendapat tugas sebuah proyek. Namun, seorang pegawai dapat libur dan tidak mendapat tugas proyek.

Deskripsi Permasalahan :

- a. Masing-masing bagian hanya mempunyai satu pengawas
- b. Seorang pengawas hanya bertugas pada satu bagian
- c. Masing-masing bagian memiliki paling tidak satu pegawai
- d. Masing-masing pegawai bekerja paling tidak pada satu bagian
- e. Masing-masing proyek dikerjakan oleh paling tidak satu pegawai
- f. Seorang Pengawas bisa mendapat tugas 0 atau beberapa proyek

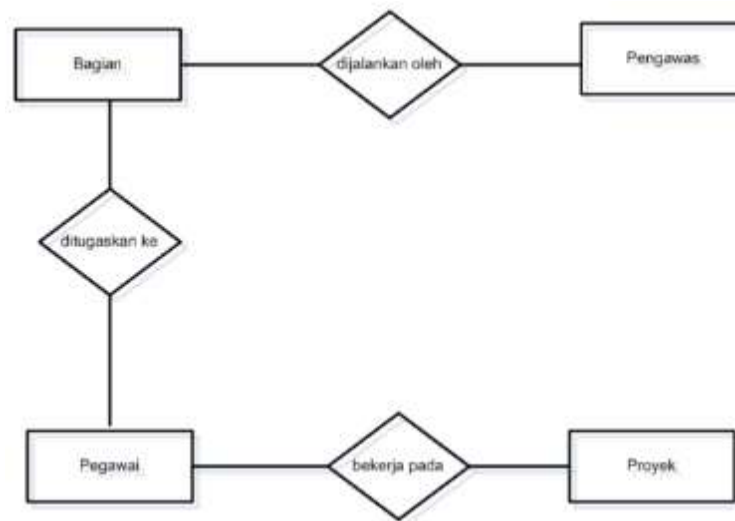
Langkah 1 : Menentukan Entitas Entitas yang dibutuhkan adalah : Bagian, Pegawai, Pengawas, dan Proyek

Langkah 2 : Menentukan Relasi dengan matriks relasi

	Bagian	Pegawai	Pengawas	Proyek
Bagian		ditugaskan ke	dijalankan oleh	
Pegawai	milik			bekerja pada
Pengawas	menjalankan			
Proyek		menggunakan		

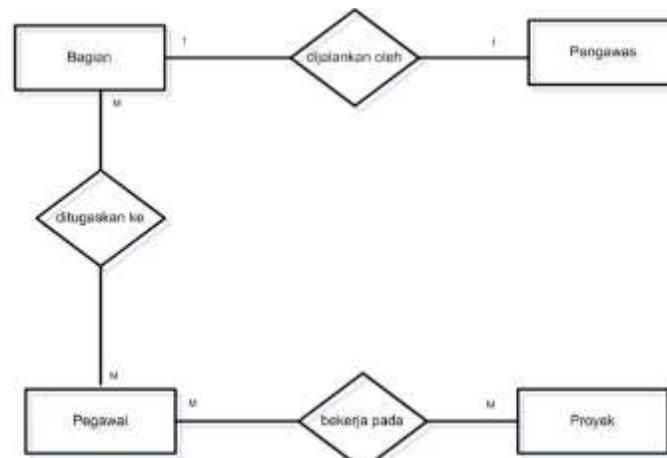
Gambar IV.18
Hasil Langkah kedua

Langkah 3 : Menggambar ERD Sementara



Gambar IV.19
Hasil Langkah ketiga

Langkah 4 : Mengisi Kardinalitas

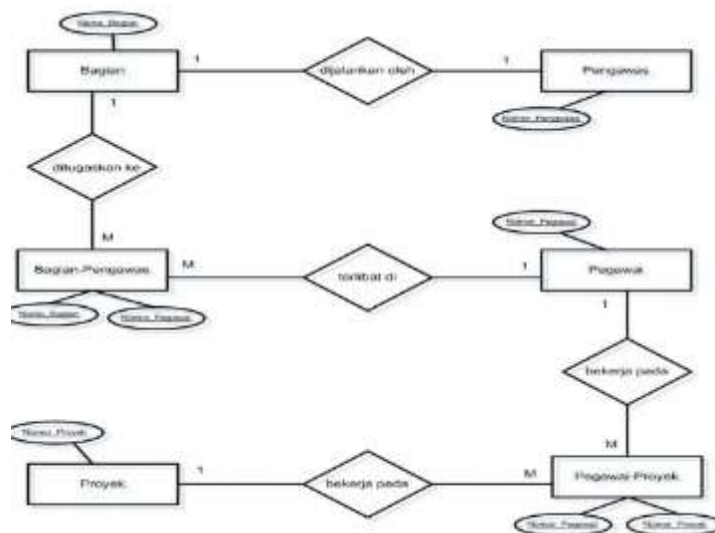


Gambar IV.20
Hasil Langkah keempat

Langkah 5 : Menentukan Kunci Utama Kunci Utama : Nama Bagian, Nomor Pengawas, Nomor Pegawai, Nomor Proyek.

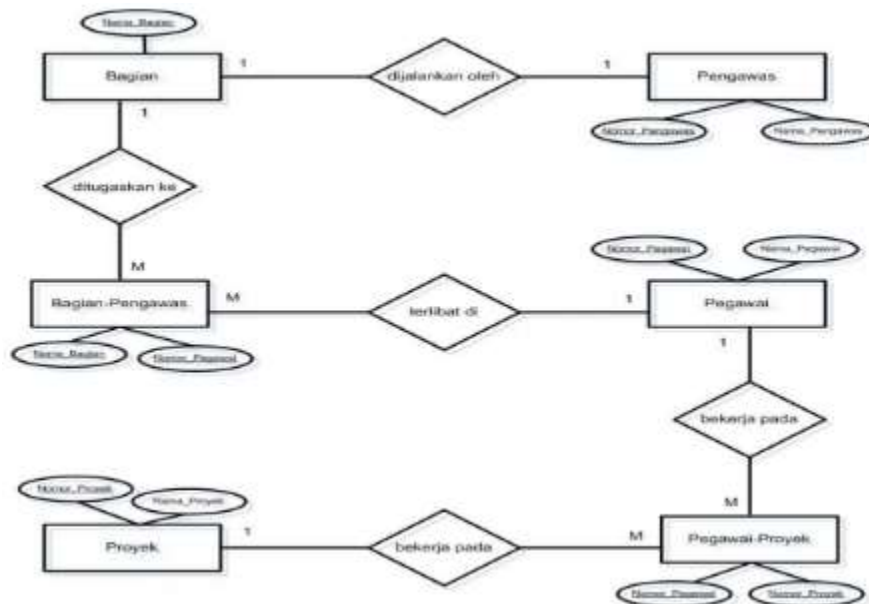
Langkah 6 : Menggambarkan ERD berdasarkan kunci

Karena ada dua relasi many-to-many pada ERD sementara, yaitu antara Bagian dan Pegawai, serta Pegawai dan Proyek. Oleh karena itu dibuatkan entitas baru yaitu Bagian-Pegawai dan Pegawai-Proyek. Kunci utama Bagian-Pegawai adalah gabungan Nama Bagian dan Nomor Pegawai. Kunci utama Pegawai-Proyek adalah gabungan Nomor Pegawai dan Nomor Proyek



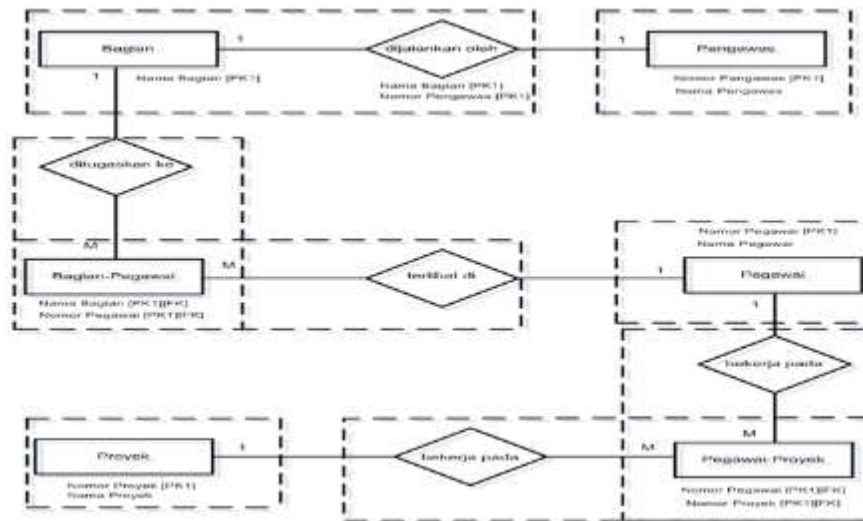
Gambar IV.21
ERD berdasarkan Kunci

Langkah 7: Menentukan Atribut yang diperlukan



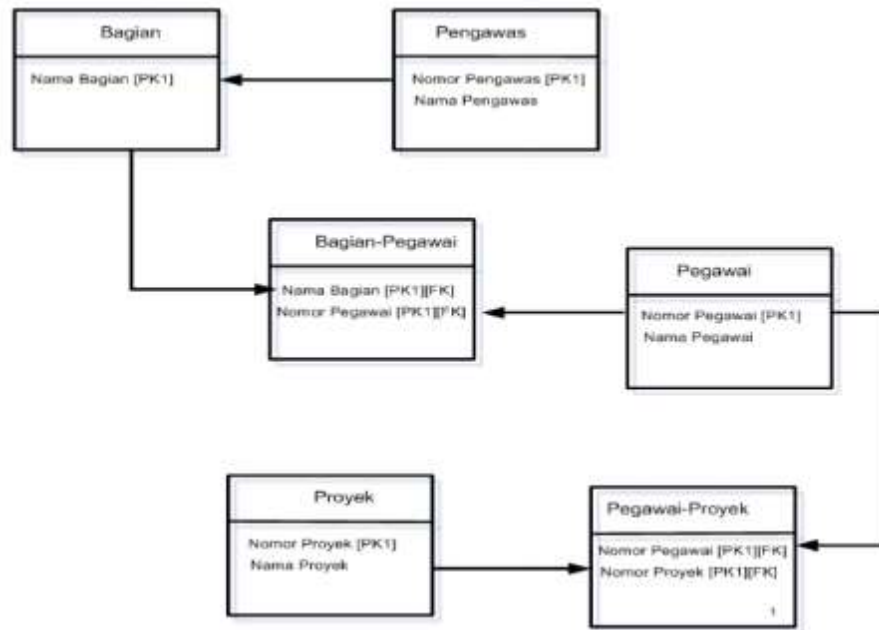
Gambar IV.22
Hasil Langkah ketujuh

Langkah 8 : Transformasi ERD ke LRS



Gambar IV.23
Transformasi ERD ke LRS

LRS Yang Terbentuk :



Gambar IV.18
LRS yang Terbentuk