

**IF2140 Pemodelan Basis Data**  
**Diagram Relasional dan Implementasi Basis Data**  
**Klik Indomaret**



**Disusun oleh:**  
**Kelompok K2-G09**

Syakira Fildza Nazhifan     / 18222012  
Axelius Davin                 / 18222016  
Erwan Poltak Halomoan     / 18222028  
Muhammad Yaafi Wasesa    / 18222052  
Monica Angela Hartono      / 18222078

**Program Studi Sistem dan Teknologi Informasi**  
**Sekolah Teknik Elektro dan Informatika**  
**Institut Teknologi Bandung**  
**2023**

# Daftar Isi

<b>Daftar Isi.....</b>	<b>1</b>
<b>1. Deskripsi Sistem.....</b>	<b>2</b>
<b>2. ER Diagram.....</b>	<b>4</b>
<b>3. Pemetaan ER Diagram menjadi Model Relasional.....</b>	<b>5</b>
3.1 Pemetaan entity menjadi relasi.....	5
a) Strong entity.....	5
b) Weak entity.....	5
3.2 Pemetaan multivalued attributes.....	5
3.3 Pemetaan derived attributes.....	5
3.4 Pemetaan relationship menjadi relasi.....	6
a) Many to many relationship.....	6
b) One to many and many to one relationship.....	6
c) One to one relationship.....	6
3.5 Pemetaan ternary relationship.....	6
3.6 Pemetaan specialization/generalization menjadi relasi.....	6
3.7 Tabel view dari derived attributes.....	6
3.8 Foreign keys.....	7
3.9 Hasil pemetaan.....	8
a) Relasi.....	8
b) Foreign keys.....	8
<b>4. Diagram Model Relasional.....</b>	<b>9</b>
<b>5. Daftar Relasi.....</b>	<b>9</b>
1. Relasi product_category.....	9
2. Relasi product.....	10
3. Relasi supplier.....	11
4. Relasi order.....	12
5. Relasi transaction.....	13
6. Relasi courier.....	13
7. Relasi review.....	14
8. Relasi user.....	15
9. Relasi member.....	16
10. Relasi guest.....	17
11. Relasi voucher.....	18
12. Relasi address.....	19
<b>Daftar Pustaka.....</b>	<b>20</b>
<b>Pembagian Tugas.....</b>	<b>21</b>
<b>Lampiran.....</b>	<b>23</b>

## 1. Deskripsi Sistem



Klik Indomaret merupakan suatu platform belanja *online* yang dimiliki oleh Indomaret, sebuah jaringan minimarket terkemuka di Indonesia. Melalui Klik Indomaret, pengguna dapat melakukan pembelian berbagai produk, termasuk makanan, minuman, barang kebutuhan sehari-hari, perlengkapan rumah tangga, dan masih banyak lagi secara *online*. Klik Indomaret juga menawarkan berbagai promosi dan diskon khusus untuk pelanggan yang berbelanja melalui platform ini.

Kami sebagai salah satu pengguna Klik Indomaret tertarik untuk menganalisisnya karena di zaman sekarang, kebanyakan orang memilih untuk melakukan sesuatu dengan cepat dan mudah. Klik Indomaret menjadi salah satu dari banyak layanan belanja *online* yang memungkinkan pelanggan untuk berbelanja dengan nyaman dan menghemat waktu tanpa harus datang ke toko fisiknya. Oleh karena itu, kami sebagai mahasiswa Sistem dan Teknologi Informasi (STI) ITB yang memperoleh ilmu mata kuliah Pemodelan Basis Data (PBD) tertarik untuk menganalisis sistem yang digunakan Klik Indomaret untuk layanan belanja *online*. Basis data yang kami analisis adalah data *supplier*, data *product*, data transaksi, data *order*, data *consumer*, dan data kurir.

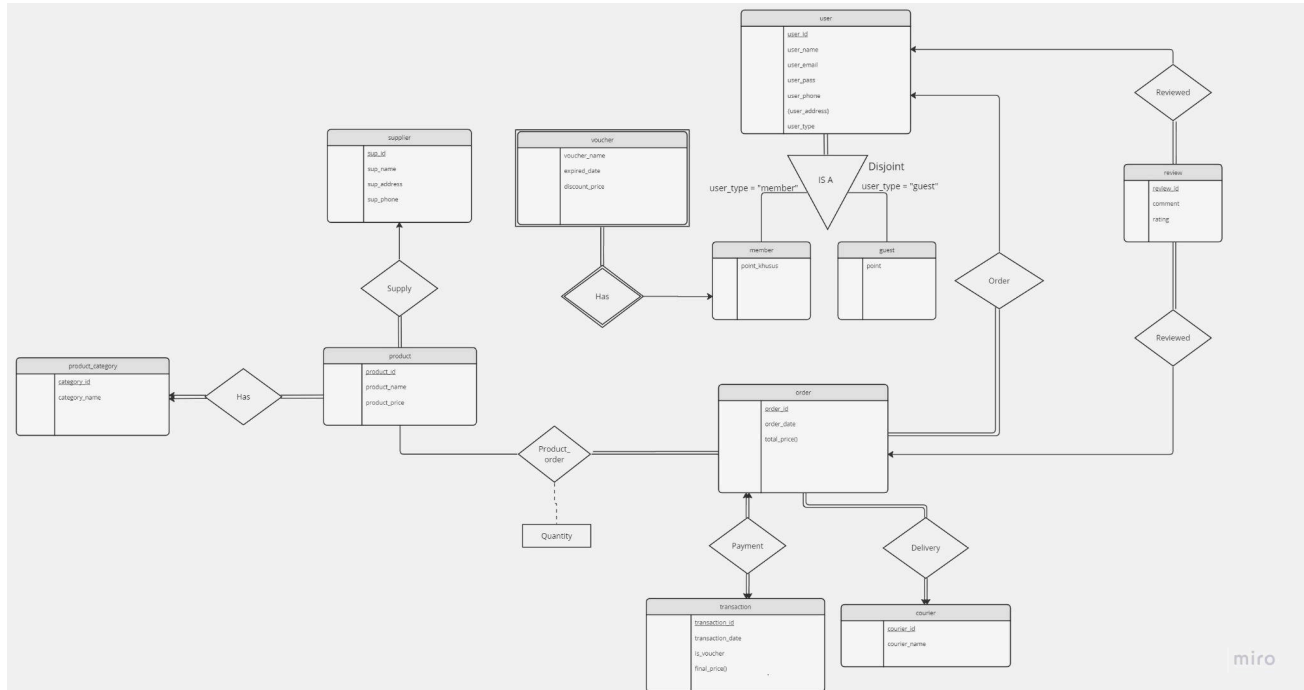
Konsumen (*user*) yang telah log in ke aplikasi dapat melakukan pemesanan (*order*). Dengan identitas konsumen yang terdiri atas ID konsumen (*user\_id*) yang unik, nama konsumen (*user\_name*), nomor telepon konsumen (*user\_phone*), alamat konsumen (*user\_address*), dan tipe konsumen (*consumer\_type*), dimana setiap konsumen akan dikategorikan sebagai *member* atau *non member*. Jika dikategorikan sebagai *member*, maka akan memiliki poin khusus yang dapat digunakan untuk membeli *voucher* (*point\_khusus*), sedangkan *guest* akan memiliki poin, namun poin tersebut tidak dapat digunakan untuk membeli *voucher*. Setiap *user* dapat memiliki banyak alamat rumah. Setiap *user* dipastikan dapat melakukan lebih dari satu *order* dan setiap *order* hanya dapat dipesan oleh 1 *user*. Setiap pemesanan (*order*) akan dicatat ID order (*order\_id*) yang unik, tanggal order (*order\_date*), dan jumlah order (*order\_quantity*).

Untuk setiap pemesanan (*order*) yang dilakukan akan terhubung ke transaksi (*transaction*). Setiap *order* dapat memiliki sejumlah *transaction* dan setiap *transaction* dapat memiliki sejumlah *order*. Setiap transaksi memiliki ID transaksi (*transaction\_id*) yang unik dan total harga yang ditentukan berdasarkan ID transaksi (*total\_price()*). Untuk setiap *order* yang dilakukan pasti akan terhubung dengan satu kurir yang akan mengantar. Setiap kurir pasti memiliki identitas yang terdiri dari nama kurir (*nama\_Kurir*), ID kurir yang unik (*id\_Kurir*). Order digabungkan dengan kurir dan setiap order berasal dari produk.

Setelah order diterima oleh *user*, *user* dapat memberikan *review*. *Review* terdiri dari ID *review* yang unik (*review\_id*), ulasan (*comment*), dan rating (*rating*). Setiap voucher memiliki nama voucher (*voucher\_name*), tanggal berlaku (*expired\_date*), dan besar diskon (*discount*). Setiap voucher pasti memiliki member dan satu member mungkin untuk mendapatkan banyak voucher.

Setiap produk yang tersedia di Klik Indomaret pasti berasal dari *supplier*. Setiap *supplier* memiliki ID *supplier* yang unik (*sup\_id*), nama *supplier* (*sup\_name*), alamat *supplier* (*sup\_address*), dan nomor telepon (*sup\_phone*). Setiap produk memiliki ID produk yang unik (*product\_id*), nama produk (*product\_name*), dan harga produk (*product\_price*). Setiap produk dapat termasuk beberapa kategori. Setiap kategori produk terdiri dari ID kategori yang unik (*category\_id*), nama kategori (*category\_name*), dan ID *supplier* (*sup\_id*).

## 2. ER Diagram



Untuk ER Diagram yang lebih jelas dapat dilihat [di sini](#)

### Asumsi :

1. Indomaret dan *user* berada di 1 kota.
2. Setiap *user* hanya memiliki 1 nomor telepon.
3. Setiap *user* dapat memiliki lebih dari 1 alamat.
4. *Voucher* hanya digunakan sebagai potongan harga.
5. Tidak ada opsi pengambilan barang langsung di toko (hanya dengan pengiriman).
6. Kurir dibayar oleh perusahaan, tidak masuk dalam perhitungan transaksi.
7. 1 jenis *product* hanya berasal dari 1 *supplier*.
8. 1 jenis *product* hanya memiliki 1 *category*.
9. *Voucher* yang sama dapat digunakan oleh banyak pengguna karena *voucher* dibeli menggunakan poin.
10. Semua order sudah dilakukan pembayaran.
11. 1 *voucher* dapat digunakan oleh banyak member dengan kuota yang sudah ditentukan.

### Hal yang tidak dapat ditulis dalam E-R:

1. Hubungan antara *voucher* dan *transaction* tidak bisa ditulis dalam E-R tetapi *voucher* bisa dipakai untuk mengurangi *total\_price* (jika punya).

### 3. Pemetaan ER Diagram menjadi Model Relasional

#### 3.1 Pemetaan *entity* menjadi relasi

##### a) Strong entity

Pemetaan *strong entity* dilakukan dengan cara membuat tabel (relasi) berisi seluruh atribut yang sama

- supplier = (sup\_id, sup\_name, sup\_address, sup\_phone)
- product = (product\_id, category\_id, sup\_id, product\_name, product\_price)
- product\_category = (category\_id, category\_name)
- order = (order\_id, order\_date, transaction\_id, courier\_id, user\_id)
- transaction = (transaction\_id, transaction\_date, is\_voucher, order\_id)
- courier = (courier\_id, courier\_name)
- review = (review\_id, comment, rating, order\_id, user\_id)
- user = (user\_id, user\_name, user\_email, user\_pass, user\_phone, user\_type, order\_id, review\_id)
- member = (user\_id, point\_khusus)
- guest = (user\_id, point)

##### b) Weak entity

Pemetaan *weak entity* dilakukan dengan cara membuat relasi yang menyertakan *primary key* dari *strong entity*. Sehingga *primary key* dari *strong entity*. Sehingga *primary key* dari *weak entity* adalah diskriminator dari *weak entity* dan *primary key* dari *strong entity*.

- voucher = (voucher\_name, user\_id, expired\_date, discount\_price)

#### 3.2 Pemetaan *multivalued attributes*

Pemetaan multivalued attributes direpresentasikan dengan membuat satu relasi baru yang berisi *primary key* dari relasi dan isi dari *multivalued attributes* itu sendiri.

- Atribut user\_adress pada entity user akan menjadi suatu relasi baru (address) dengan *primary key*-nya adalah *primary key* dari user, yakni user\_id dan user\_address.  
address = (user\_id, user\_address)

#### 3.3 Pemetaan *derived attributes*

*Derived attributes* tidak akan diimplementasikan dalam model relasional dan diabaikan. *Derived attributes* akan direpresentasikan dalam bentuk view.

- Atribut total\_price pada entity order dihapus ketika membuat relasi order

### 3.4 Pemetaan *relationship* menjadi relasi

#### a) Many to many relationship

Pemetaan *relationship many to many* dilakukan dengan membuat relasi baru dengan *primary key*-nya merupakan hasil penggabungan dari *primary key* kedua *entity*.

- product\_order = (product\_id, order\_id)

#### b) One to many and many to one relationship

Pemetaan *relationship one to many* dan *many to one* dilakukan dengan menambahkan atribut *primary key* dari bagian “one” ke bagian “many”.

- product = (product\_id, category\_id, sup\_id, product\_name, product price)
- order = (order\_id, user\_id, courier\_id, order\_date)
- review = (review\_id, user\_id, order\_id, comment, rating)

#### c) One to one relationship

Pemetaan *relationship one to one* dilakukan dengan menambahkan atribut *primary key* pada salah satu *entity* ke *entity* lain. Contohnya:

- Dalam *relationship payment*, relasi order akan mendapatkan tambahan atribut transaction\_id dari relasi transaction dan relasi transaction akan mendapatkan tambahan atribut order\_id dari relasi order.

### 3.5 Pemetaan *ternary relationship*

Tidak ada *ternary relationship* pada model ER diagram kami

### 3.6 Pemetaan *specialization/generalization* menjadi relasi

Pemetaan *specialization/generalization* dapat dilakukan dengan membuat 3 relasi, relasi pertama adalah semua atribut User, relasi kedua adalah semua atribut User atau *primary key*nya saja ditambah atribut dari entity member, relasi ketiga adalah semua atribut User atau *primary key*nya saja ditambah atribut dari entity guest.

### 3.7 Tabel view dari *derived attributes*

#### a) OrderPrice()

Tabel view OrderPrice() merupakan tabel yang berasal dari relasi *order* dan menjelaskan total biaya yang terhitung dari jumlah harga barang yang dipesan.

<u>order_id</u>	total_price
O089	91000
O137	147000

O139	122000
------	--------

b) FinalPrice()

Tabel view OrderPrice() merupakan tabel yang berasal dari relasi *order* dan menjelaskan total biaya akhir yang telah dibayarkan oleh *user* setelah menggunakan *voucher*/tidak.

<u>transaction_id</u>	final_price
O089	91000
O137	132000
O139	117000

### 3.8 Foreign keys

Pemetaan *foreign keys* bisa dilihat dari setiap entity yang diambil dari *primary key* entity lain. Foreign key dapat terbentuk karena

- Adanya weak entity
  - voucher(voucher\_name) → member(member\_id)
- Adanya multivalued attribute
  - user-user\_address(user\_id) → user(user\_id)
- Pembuatan relasi baru akibat many to many relationship
  - product\_order(product\_id) → product(product\_id)
  - product\_order(order\_id) → order(order\_id)
- Penyisipan atribut akibat *one to many relationship* atau *many to one relationship*
  - product(category\_id) → product-category(category\_id)
  - product(sup\_id) → supplier(sup\_id)
  - order(user\_id) → user(user\_id)
  - order(courier\_id) → courier(courier\_id)
- Penyisipan atribut akibat one to one relationship
  - transaction(order\_id) → order(order\_id)
  - order(user\_id) → user(user\_id)
  - review(order\_id) → order(order\_id)
- Akibat adanya *specialization / generalization*
  - user-member(user\_id) → user(user\_id)
  - user-guest(guest\_id) → user(user\_id)



### 3.9 Hasil pemetaan

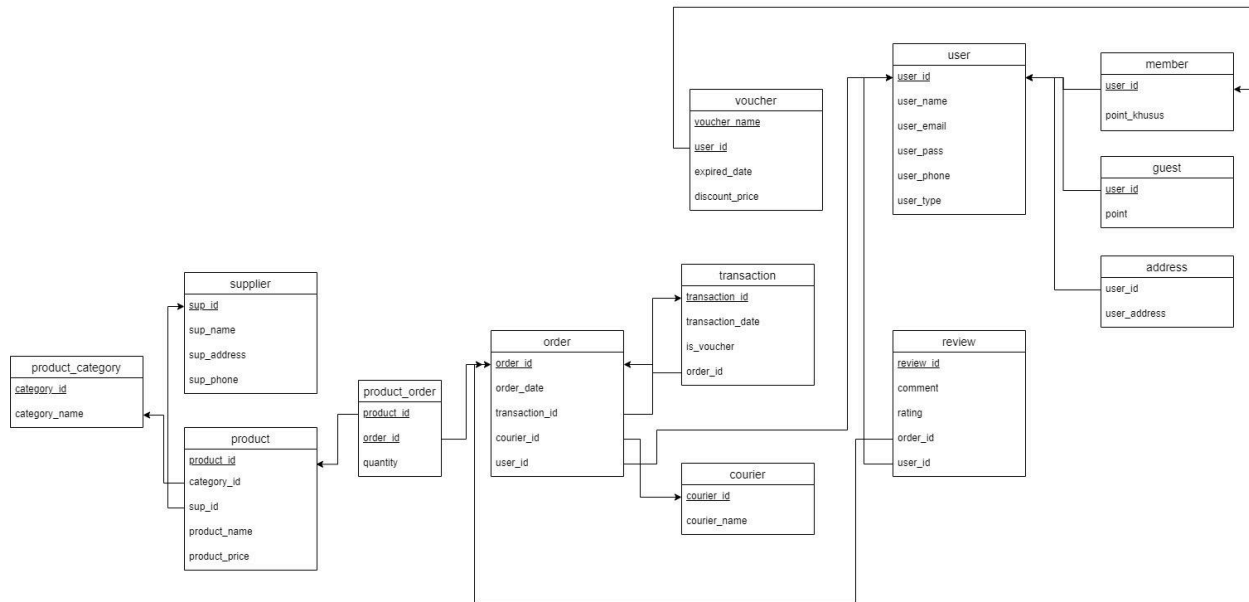
#### a) Relasi

- product\_category = (category\_id, category\_name)
- product = (product\_id, category\_id, sup\_id, product\_name, product\_price)
- supplier = (sup\_id, sup\_name, sup\_address, sup\_phone)
- order = (order\_id, order\_date, transaction\_id, courier\_id, user\_id)
- transaction = (transaction\_id, transaction\_date, is\_voucher, order\_id)
- courier = (courier\_id, courier\_name)
- review = (review\_id, comment, rating, order\_id, user\_id)
- user = (user\_id, user\_name, user\_email, user\_pass, user\_phone, user\_type, order\_id, review\_id)
- member = (user\_id, point\_khusus)
- guest = (user\_id, point)
- voucher = (voucher\_name, expired\_date, discount\_price)

#### b) Foreign keys

- voucher(voucher\_name) → member(member\_id)
- user-user\_address(user\_id) → user(user\_id)
- product\_order(product\_id) → product(product\_id)
- product\_order(order\_id) → order(order\_id)
- product(category\_id) → product-category(category\_id)
- product(sup\_id) → supplier(sup\_id)
- order(user\_id) → user(user\_id)
- order(courier\_id) → courier(courier\_id)
- transaction(order\_id) → order(order\_id)
- review(order\_id) → order(order\_id)
- review(user\_id) → user(user\_id)
- user-member(user\_id) → user(user\_id)
- user-guest(guest\_id) → user(user\_id)

## 4. Diagram Model Relasional



## 5. Daftar Relasi

## 1. Relasi product\_category

```
product_category = (category_id, category_name)
```

FK:

—

Deskripsi:

1. `category_id` merupakan nomor ID yang dimiliki oleh masing-masing *category*.
2. `category_name` adalah nama kategori untuk setiap ID *category*.

Keterangan:

—

Field	Type	Null	Key
category_id	VARCHAR(4)	NO	PK
category_name	VARCHAR(19)	NO	

<u>category_id</u>	category_name
C001	pet supplies
C002	produce

C003	meet
C004	canned goods
C005	frozen food

## 2. Relasi product

product = (product\_id, category\_id, sup\_id, product\_name, product\_price)

FK:

- product(sup\_id) → supplier(sup\_id)
- product(category\_id) → product-category(category\_id)

Deskripsi:

1. product\_id merupakan nomor ID yang dimiliki oleh masing-masing *product*.
2. category\_id merupakan nomor ID yang dimiliki oleh masing-masing *category*.
3. sup\_id merupakan nomor ID yang dimiliki oleh masing-masing *supplier*.
4. product\_name merupakan nama dari masing-masing *product*.
5. product\_price merupakan harga dari masing-masing *product*.

Keterangan:

-

Field	Type	Null	Key
product_id	VARCHAR(4)	NO	PRI
category_id	VARCHAR(4)	NO	FK
sup_id	VARCHAR(4)	NO	FK
product_name	VARCHAR(32)	NO	
product_price	INT	NO	

<u>product_id</u>	category_id	sup_id	product_name	product_price
P001	C048	S137	Coconut and Lime Shower Gel	48000
P002	C135	S095	Pineapple and	88000

			Coconut Lip Balm	
P003	C115	S030	Mixed Nuts Variety Pack	37000
P004	C063	S057	Vanilla Bean Hand Cream	71000
P005	C095	S048	Salted Caramel Protein Bar	58000

### 3. Relasi supplier

supplier = (sup\_id, sup\_name, sup\_address, sup\_phone)

FK:

-

Deskripsi:

1. sup\_id merupakan nomor ID yang dimiliki oleh masing-masing *supplier*.
2. sup\_name merupakan nama dari *supplier*.
3. sup\_address merupakan alamat dari *supplier*.
4. sup\_phone merupakan nomor telepon dari *supplier*.

Keterangan:

-

Field	Type	Null	Key
sup_id	VARCHAR(4)	NO	PK
sup_name	VARCHAR(20)	NO	
sup_address	VARCHAR(39)	NO	
sup_phone	VARCHAR(15)	NO	

<u>sup_id</u>	sup_name	sup_address	sup_phone
S001	Malorie Bignell	Jl. Flamboyan No. 27. Makassar	087890123460
S002	Joannes Robrose	Jl. Kesawan No. 8. Medan	081234567891
S003	Nady Cressor	Jl. Raya Barat No. 5. Medan	086789012354

S004	Corrine Longcaster	Jl. Sengkaling No. 3. Malang	084567890131
S005	Chaunce Linscott	Jl. Ketintang No. 17. Surabaya	088901234580

#### 4. Relasi order

Order = (order\_id, order\_date, transaction\_id, courier\_id, user\_id)

FK:

- order(transaction\_id) → transaction(transaction\_id)
- order(user\_id) → user(user\_id)
- order(courier\_id) → courier(courier\_id)

Deskripsi:

1. order\_id adalah nomor ID pada setiap *order*.
2. order\_date adalah tanggal user melakukan *order* (YYYY-MM-DD).
3. transaction\_id adalah nomor ID pada setiap transaksi.
4. courier\_id adalah nomor ID yang dimiliki oleh setiap kurir.
5. user\_id adalah nomor ID yang dimiliki oleh setiap *user*.

Keterangan:

-

Field	Type	Null	Key
order_id	VARCHAR(4)	NO	PK
order_date	DATE	NO	
transaction_id	VARCHAR(4)	NO	FK
courier_id	VARCHAR(4)	NO	FK
user_id	VARCHAR(4)	NO	FK

<u>order_id</u>	order_date	transaction_id	courier_id	user_id
O001	2023-07-27	T001	K001	U001
O002	2022-11-22	T002	K002	U002
O003	2023-01-01	T003	K003	U003
O004	2022-11-17	T004	K004	U004

O005	2023-09-16	T005	K005	U005
------	------------	------	------	------

##### 5. Relasi transaction

transaction = (transaction\_id, transaction\_date, is\_voucher, order\_id)

FK:

- transaction(order\_id) → order(order\_id)

Deskripsi:

1. transaction\_id merupakan nomor ID pada setiap transaksi.
2. transaction\_date merupakan tanggal saat melakukan transaksi (YYYY-MM-DD).
3. id\_voucher digunakan untuk mengetahui apakah transaksi yang dilakukan menggunakan *voucher* atau tidak.
4. order\_id merupakan nomor ID pada setiap *order*.

Keterangan:

-

Field	Type	Null	Key
transaction_id	VARCHAR(4)	NO	PK
transaction_date	DATE	NO	
is_voucher	VARCHAR(5)	NO	
order_id	VARCHAR(4)	NO	FK

<u>transaction_id</u>	transaction_date	is_voucher	order_id
T001	2023-11-05	FALSE	O001
T002	2023-11-07	TRUE	O002
T003	2023-11-04	TRUE	O003
T004	2023-11-06	FALSE	O004
T005	2023-11-10	FALSE	O005

##### 6. Relasi courier

courier = (courier\_id, courier\_name)

FK:

-

Deskripsi:

1. courier\_id adalah nomor ID yang dimiliki oleh setiap kurir
2. courier\_name adalah nama dari setiap kurir.

Keterangan:

-

Field	Type	Null	Key
courier_id	VARCHAR(4)	NO	PK
courier_name	VARCHAR(10)	NO	

<u>courier_id</u>	courier_name
K001	Tailor
K002	Farrel
K003	Anders
K004	Parren
K005	Isidor

## 7. Relasi review

review = (review\_id, comment, rating, order\_id, user\_id)

FK:

- review(order\_id) → order(order\_id)
- review(user\_id) → user(user\_id)

Deskripsi:

1. review\_id merupakan ID dari *review* yang diberikan.
2. comment merupakan tanggapan yang diberikan oleh *user*.
3. rating merupakan penilaian yang diberikan oleh *user*.
4. order\_id merupakan ID dari masing-masing *order*.
5. user\_id merupakan ID dari masing-masing *user*.

Keterangan:

- Jumlah data pada tabel *review* tidak lebih dari 150, karena tidak semua *order* memiliki *review*.
- Apabila *user* tidak melakukan rating, maka data tidak akan masuk ke dalam tabel, namun apabila *user* melakukan rating, namun tidak memberi *comment*, maka *comment* akan Null

Field	Type	Null	Key
review_id	VARCHAR(4)	NO	PK
comment	VARCHAR(34)	YES	
rating	INT	NO	
order_id	VARCHAR(4)	NO	FK
user_id	VARCHAR(4)	NO	FK

<u>review_id</u>	comment	rating	order_id	user_id
R001	NULL	2	O105	U119
R002	Terrible quality. Do not buy.	5	O074	U142
R003	NULL	5	O003	U122
R004	NULL	3	O031	U017
R005	Fast and easy shopping experience.	1	O026	U050

#### 8. Relasi user

user = (user\_id, user\_name, user\_email, user\_pass, user\_phone, user\_type, order\_id, review\_id)

FK:

-

Deskripsi:

1. user\_id merupakan ID dari masing-masing *user*.
2. user\_name merupakan nama dari masing-masing *user*.
3. user\_email merupakan *email* dari masing-masing *user*.
4. user\_pass merupakan *password* dari masing-masing *user*.
5. user\_phone merupakan nomor telepon dari masing-masing *user*.
6. user\_type merupakan tipe dari *user* (*member* atau *guest*).

Keterangan:

-



Field	Type	Null	Key
user_id	VARCHAR(4)	NO	PK
user_name	VARCHAR(12)	NO	
user_email	VARCHAR(32)	NO	
user_pass	VARCHAR(16)	NO	
user_phone	VARCHAR(15)	NO	
user_type	VARCHAR(6)	NO	
order_id	VARCHAR(4)	NO	FK
review_id	VARCHAR(4)	NO	FK

<u>user_id</u>	user_name	user_email	user_pass	user_phone	user_type
U001	Kleinzweig	ckleinzweig0@google.nl	"vV9@_<Sly\$""cP`\$"	085678901239	member
U002	Brecknall	gbrecknall1@sohu.com	"fJ5)J`v3Zl'CX\X""	085678901242	member
U003	Oxbury	foxbury2@google.com.au	oL9.5GB!.3J*\$	082345678904	member
U004	Hanne	thanne3@phpbb.com	mB8% w Y	086789012356	member
U005	Gariff	egariff4@springer.com	xD1)4Lt'J	089012345679	member

## 9. Relasi member

member = (user\_id, point\_khusus)

FK:

- member(user\_id) → user(user\_id)

Deskripsi:

1. user\_id adalah nomor ID yang dimiliki oleh setiap *user*.
2. point\_khusus adalah jumlah *point* yang dimiliki oleh *member*.

Keterangan:

- point\_khusus dapat digunakan untuk membeli *voucher*.

Field	Type	Null	Key
user_id	VARCHAR(4)	NO	PK
point_khusus	INT	NO	

<u>user_id</u>	point_khusus
U001	8485
U002	1011
U003	10000
U004	4647
U005	9495

#### 10. Relasi guest

guest = (user\_id, point)

FK:

- guest(user\_id) → user(user\_id)

Deskripsi:

1. user\_id adalah nomor ID yang dimiliki oleh setiap *user*.
2. point adalah total point yang dimiliki oleh *guest*.

Keterangan:

- Point pada *guest* tidak bisa digunakan untuk membeli *voucher*.

Field	Type	Null	Key
user_id	VARCHAR(4)	NO	PK
point	INT	NO	

<u>user_id</u>	point
U076	6465
U077	6465
U078	9495

U079	1617
U080	2223

#### 11. Relasi voucher

voucher = (voucher\_name, user\_id, expired\_date, discount\_price)

FK:

- voucher(user\_id) → user(user\_id)

Deskripsi:

1. voucher\_name adalah nama *voucher*.
2. expired\_date adalah tanggal *voucher* tidak berlaku lagi.
3. discount\_price adalah besar harga diskon pada *voucher*.

Keterangan:

- discount\_price hanya memiliki tiga nominal, yaitu Rp 5.000, Rp. 10.000, Rp 15.000.

Field	Type	Null	Key
voucher_name	VARCHAR(15)	NO	PK
user_id	VARCHAR(4)	NO	PK
expired_date	DATE	NO	
discount_price	INT	NO	

<u>voucher_name</u>	<u>user_id</u>	expired_date	discount_price
PromoPinnacle	U069	2023-02-25	15000
PromoPanache	U058	2023-01-29	5000
BargainBinge	U007	2023-03-15	15000
DiscountDeluge	U073	2022-12-11	10000
VoucherVista	U016	2023-05-27	10000

#### 12. Relasi address

address = (user\_id, user\_address)

FK:

-

Deskripsi:

- user\_id merupakan ID dari masing-masing *user*.
- user\_address merupakan alamat dari masing-masing *user*.

Keterangan:

- Relasi address merupakan relasi dari *multivalued attribute* dengan anggapan satu *user* dapat menempati beberapa alamat.

Field	Type	Null	Key
user_id	VARCHAR(4)	NO	FK
user_address	VARCHAR(42)	NO	

user_id	user_address
U001	Jl. Thamrin No. 59595. Ende
U002	Jl. Diponegoro No. 40404. Gunungsitoli
U003	Jl. Veteran No. 7777. Padang
U004	Jl. Gajah Mada No. 27272. Tegal
U005	Jl. Diponegoro No. 28282. Serang

### 13. Relasi product\_order

product\_order = (product\_id, order\_id, quantity)

FK:

- product\_order(product\_id) → product(product\_id)
- product\_order(order\_id) → order(order\_id)

Deskripsi:

- product\_id merupakan ID dari masing-masing produk.
- order\_id merupakan ID dari masing-masing order.
- quantity merupakan banyaknya produk dalam setiap order.

Keterangan:

-

Field	Type	Null	Key
product_id	VARCHAR(4)	NO	PK, FK

order_id	VARCHAR(4)	NO	PK, FK
quantity	INT	NO	

<u>product_id</u>	<u>order_id</u>	quantity
P061	O036	7
P149	O076	2
P142	O124	6
P032	O096	6
P064	O105	10

## **Daftar Pustaka**

A. Silberschatz, H.F. Korth, S. Sudarshan, 'Database System Concepts', McGraw-Hill

J.A. Hoffer, M.B. Prescott, F.R. McFadden, 'Modern Database Management', Pearson Prentice Hall

## Pembagian Tugas

Nama Lengkap - NIM	Deskripsi Tugas
Syakira Fildza Nazhifan - 18222012	<ul style="list-style-type: none"> <li>- Melakukan pembagian tugas dan set timeline pengerjaan.</li> <li>- Membuat pemetaan <i>strong entity</i> dan <i>weak entity</i> menjadi relasi.</li> <li>- Membuat diagram model relasional.</li> <li>- Membuat daftar relasi <i>order</i>, <i>voucher</i>, <i>guest</i>, dan <i>member</i>.</li> <li>- Merapikan dokumen.</li> </ul>
Axelius Davin - 18222016	<ul style="list-style-type: none"> <li>- Membuat pemetaan <i>specialization/generalization</i> menjadi relasi.</li> <li>- Membuat hasil pemetaan foreign keys.</li> <li>- Membuat diagram model relasional.</li> <li>- Melakukan <i>generate data dummy</i>.</li> <li>- Membuat data sql.</li> </ul>
Erwan Poltak Halomoan - 18222028	<ul style="list-style-type: none"> <li>- Membuat pemetaan <i>multivalued attributes</i> menjadi relasi.</li> <li>- Membuat penjelasan pemetaan <i>derive attributes</i>.</li> <li>- Membuat <i>table view</i> dari <i>derive attributes</i>.</li> <li>- Membuat daftar relasi <i>address</i>.</li> <li>- Membuat diagram model relasional.</li> <li>- Revisi dan finalisasi pemetaan model relasional.</li> </ul>
Muhammad Yaafi Wasesa Putra - 18222052	<ul style="list-style-type: none"> <li>- Membuat pemetaan hasil pemetaan <i>foreign key</i></li> <li>- Membuat diagram model relasional</li> <li>- Membuat daftar relasi <i>product_category</i> dan <i>courier</i>.</li> <li>- Membuat daftar relasi address</li> </ul>
Monica Angela Hartono - 18222078	<ul style="list-style-type: none"> <li>- Melakukan pembagian tugas dan set timeline pengerjaan.</li> <li>- Membuat pemetaan <i>relationship</i> menjadi relasi untuk <i>many to many</i>, <i>one to many</i>, <i>one to one</i>.</li> <li>- Membuat hasil pemetaan bagian relasi.</li> </ul>

	<ul style="list-style-type: none"> <li>- Notulen asistensi dan mengisi MoM.</li> <li>- Membuat dan finalisasi diagram model relasional.</li> <li>- Melakukan <i>generate data dummy</i>.</li> <li>- Membuat daftar relasi <i>product, supplier, review, transaction, user</i>.</li> <li>- Merapikan dokumen.</li> </ul>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## Lampiran

### Moments of Meeting

#### Asistensi Milestone 2

Hari	Jumat	Tanggal	10 November 2023
Kelas	K02	Kelompok	G09
Waktu	15.00	Tempat	Zoom
Anggota Kelompok	NIM	Nama	
	18222012	Syakira Fildza Nazhifan	
	18222016	Axelius Davin	
	18222028	Erwan Poltak Halomoan	
	18222052	Muhammad Yaafi Wasesa	
	18222078	Monica Angela Hartono	
Nama Asisten	Reinhart Lim		

Hasil Asistensi		
No.	Pertanyaan	Jawaban
1.	Nama tabelnya udah oke “ <i>has</i> ” atau belum?	Sebaiknya jangan <i>has</i> , karena agak ga jelas pas udah masuk basis datanya nanti. Mending ganti jadi “ <i>product_order</i> ”.
2.	Kalo isi <i>dummy voucher_name</i> cuma potongan pembelian sama potongan ongkir aja oke ga?	Bisa ditambah-tambahin yang lain sih, ga sesuai sama Klik Indomaret nya banget gapapa, asal masih nyambung.
3.	Isi datanya 150 per atribut atau 150 per tabel?	Nantinya minimal 150 baris harus keisi, berlaku buat tabel-tabel yang mungkin aja, yang ga mungkin gapapa kalo kurang, tapi ditulis di laporan

		alasannya kenapa kurang.
4.	Cara masukin <i>dummy</i> ke SQL nya gimana?	Waktu tahun lalu, Kak Lim beberapa <i>generate</i> di Mockaroo, bisa juga convert ke csv terus dicek udah <i>unique</i> semua atau belum. Lalu, dari csv di- <i>convert</i> ke SQL pake <i>converter online</i> . (dBeaver, dll)
5.	Gimana cara ngebedain <i>point</i> di <i>member</i> dan di <i>guest</i> ?	Buat di <i>member</i> , pointnya bisa <i>point bonus</i> (ada <i>benefit</i> lebih), di <i>guest</i> cuma <i>point</i> biasa.

